

**UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITECNICA SUPERIOR**

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA



PROYECTO FIN DE CARRERA

INGENIERÍA INDUSTRIAL

***PROBABILIDAD DE SOBRECARGAS EN
LÍNEAS DE TRANSPORTE DEBIDAS A LA
INCERTIDUMBRE DE LA PRODUCCIÓN
EÓLICA***

AUTOR: Javier Vaquerizo Virto

TUTOR: Julio Usaola García

Leganés, 1 de septiembre de 2009

ÍNDICE

1. Introducción.	5
2. Situación actual en España del sistema eléctrico, la generación eólica y la incertidumbre en las previsiones.	6
3. Análisis de la incertidumbre de las previsiones de generación eólica.	12
4. Descripción y nociones básicas del programa, PSS/E.	14
5. Descripción del flujo de cargas probabilístico.	17
5.1 El método de Monte Carlo.	19
5.1.1 Principios básicos.	19
5.1.2 El teorema central del límite.	20
5.1.3 Análisis de convergencia del método.	21
5.2. Modelado del comportamiento de la generación eólica.	23
5.3. Modelado del comportamiento de las cargas.	26
5.4 Descripción del proceso para la resolución de un flujo de cargas probabilístico mediante el método de Monte Carlo	27
6. Descripción de los programas creados para la realización de un flujo de cargas probabilístico, mediante el método de Monte Carlo	30
6.1 Descripción del archivo: “Datos de Entrada”.	31
6.2 Descripción del programa: “Lectura y Almacenamiento de P y Pmax” .	32
6.3 Descripción del programa: “DBD” .	33
6.4 Descripción del programa “Adapta datos a PSS/E”.	34
6.5 Descripción del programa “FCP” , flujo de cargas probabilístico.	34
7. Cálculo del número de iteraciones mínimo.	35
8. Resultados del estudio, mediante flujo de cargas probabilístico.	39
9. Conclusiones.	53
Anexo.	57
A.1. Pasos a seguir por el usuario para ejecutar un flujo de cargas probabilístico.	57
A.2. Programas.	62
A.2.1 Descripción detallada del formato del archivo: “Datos de Entrada”.	62
A.2.2 Descripción detallada del código del programa: “Lectura y Almacenamiento de P y Pmax”	65
A.2.3 Descripción detallada del código del programa: “DBD”	70
A.2.4 Descripción detallada del código del programa “Adapta datos a PSS/E”.	75
A.2.5 Descripción detallada de código del programa “FCP”.	78
Referencias bibliográficas.	90

Resumen del proyecto.

La finalidad de este trabajo es la evaluación de la probabilidad de sobrecargas en la red de transporte de electricidad, debidas a la generación eólica. Para obtener la probabilidad de sobrecargas en las líneas de transporte y los valores más probables de tensión en las subestaciones se utilizará el método de flujo de cargas probabilístico resuelto mediante el método de Monte Carlo.

El proceso de cálculo se hará con la ayuda de MATLAB [5] y del programa de simulación de sistemas eléctricos: PSS/E, versión 31, desarrollado por SIEMENS. [4]

Se ha elegido este programa porque permite trabajar de forma sencilla e intuitiva con diferentes modelos de redes eléctricas y admite la posibilidad de programar en lenguaje Python [3], que al ser un lenguaje de programación genérico y ampliamente difundido facilita la modificación del código por parte de cualquier usuario.

En la resolución del flujo de cargas probabilístico mediante el método de Monte Carlo [2], las variables que se consideran son: producción eólica y potencia demandada. Estas variables son aleatorias, dependientes, continuas y tienen una cierta incertidumbre. [1]

Ha de considerarse la dependencia entre las potencias generadas por los parques eólicos de una misma zona geográfica, así como la incertidumbre de la potencia consumida por cada carga del sistema.

El flujo de cargas probabilístico resuelto mediante el método de Monte Carlo [2], se basa en la generación aleatoria de valores de potencia eólica producida por cada parque, según una FDP beta [10], a partir de los valores de producción previstos para cada uno de los parques eólicos del sistema y en la generación aleatoria de valores de potencia consumida por las diferentes cargas del sistema, según una FDP normal [10], a partir de las previsiones de consumo de cada carga del sistema. Los flujos de cargas se resolverán en corriente alterna.

Se parte del modelo de red eléctrica a estudiar, dicho modelo es la representación del sistema eléctrico (líneas, subestaciones, transformadores, generadores y cargas) y está contenido en un archivo del programa PSS/E [4] llamado “caso”, en el cual figuran las previsiones de generación eólica y de demanda, y los valores generación convencional programados para el escenario horario a estudiar.

A partir de los valores de potencia eólica prevista para cada parque se generan aleatoriamente, según una FDP beta y una matriz de correlación entre las potencias de los parques de una misma zona geográfica, nuevas potencias generadas por los parques eólicos y mediante una FDP normal, a partir de los valores de potencia demandada previstos, las nuevas potencias consumidas por las cargas. Se introducen los valores obtenidos en el caso de partida y se distribuye la variación de generación eólica aparecida entre los generadores convencionales. Se realiza un flujo de cargas en corriente alterna y se almacenan los valores de carga por las líneas y las tensiones en las subestaciones, que se hayan definido como objeto de estudio; para posteriormente ser comparados y estudiados.

Utilizando el método de Monte Carlo para resolver el flujo de cargas probabilístico se aproxima el resultado de la simulación a la realidad. El proceso anterior se repite un número determinado de veces (500, 1000) que habrá que determinar [6] y se estudian los valores obtenidos de: carga por las líneas y tensiones en las subestaciones. De forma que mediante un análisis en frecuencia de dichos valores se evalúa la probabilidad de sobrecargas en las líneas estudiadas y se obtienen los valores más probables de tensión en las subestaciones.

1. Introducción.

Actualmente en la operación de los sistemas eléctricos aparecen dos problemas principales causados por la generación eólica.

Uno de ellos es la pérdida de generación eólica, debida a la desconexión de la red eléctrica de los aerogeneradores, a causa de los huecos de tensión producidos por perturbaciones en la red. Éste se está solventando a medida que se consigue que la electrónica de potencia de los aerogeneradores de los parques eólicos soporte las sobreintensidades que se producen en las máquinas con los huecos de tensión; de esta forma se evita la desconexión de la red de los aerogeneradores para evitar dañar sus equipos.

El otro es la incertidumbre en las previsiones de generación eólica, que obliga a tener reservas de generación convencional para cubrir dicha incertidumbre e introduce riesgos en el sistema eléctrico. Debido a esta incertidumbre en ocasiones es necesario aumentar o disminuir la generación convencional, en la operación en tiempo real, ya que las previsiones de generación eólica no se cumplen. También aparecen en la operación en tiempo real sobrecargas en determinadas líneas, que no se previeron al realizar el análisis de restricciones el día anterior con los valores de generación eólica previstos, causadas por unos valores de generación eólica superiores a las previsiones con las que se realizó dicho análisis de restricciones.

Las previsiones de generación eólica cada vez son mejores por lo que la incertidumbre disminuye, pero aún así es necesario seguir desarrollando nuevos métodos y herramientas que consigan previsiones más exactas y ayuden en la operación de los sistemas eléctricos.

Como se dijo anteriormente, la determinación de la probabilidad de sobrecargas en las líneas de transporte eléctrico, ocasionadas por la variación de la generación eólica; se ha hecho mediante un flujo de cargas probabilístico resuelto mediante el método de Monte Carlo [2]. Este método parte de los valores previstos de generación eólica y potencia demandada. Genera mediante las funciones de densidad de probabilidad (FDP) de las dos variables consideradas (producción eólica y potencia demandada) valores aleatorios de potencia generada y consumida por los parques eólicos y las cargas, respectivamente; introduce dichos valores en el modelo de red con el que se está trabajando, realiza un flujo de cargas y almacena los valores de carga de las líneas y tensiones en las subestaciones. Este proceso se repite un número determinado de veces y una vez concluido se realiza un análisis en frecuencia de los valores de carga y tensiones almacenados, obteniendo así la probabilidad de sobrecarga de las líneas y los valores más probables de tensión en las subestaciones.

2. Situación actual en España del sistema eléctrico, la generación eólica y la incertidumbre en las previsiones.

Cada vez más la generación eólica juega un papel muy importante en determinados sistemas eléctricos, ya que la potencia eólica instalada es un porcentaje importante de la generación total y aporta buena parte de la potencia demandada. Sin embargo, como se dijo anteriormente, esta generación eólica introduce complicaciones en la operación de los sistemas eléctricos debido a la incertidumbre en la previsión de su potencia generada y a la sensibilidad de los aerogeneradores frente a perturbaciones en la red eléctrica.

En este momento, a 1 de septiembre de 2009, se han superado los 17.000 MW de potencia eólica instalada en el sistema eléctrico español, lo que supone alrededor de un 20% de la potencia total instalada en el sistema eléctrico español. Se trata de una tecnología que tiene una producción muy variable y una previsión relativamente difícil. De hecho se puede afirmar que es la segunda consumidora de reservas de generación del sistema eléctrico español. Cualquier central en España puede fallar en cualquier momento y hay que tener reservas suficientes para cubrir ese fallo.

El fallo fortuito de las centrales de régimen ordinario es el primer consumidor de reservas para poder operar el sistema eléctrico de forma segura. Y el segundo consumidor, en este momento, es la variabilidad de la producción eólica. El margen de incertidumbre que deja su previsión para un horizonte entre 12 y 36 horas, en estos momentos se sitúa en torno a unos 600 MW de reserva caliente necesaria para cubrir esa variabilidad que introducen estos, más de 17.000 MW eólicos que hay instalados en el sistema español. Entendiendo por reserva caliente o reserva terciaria, el margen de potencia a subir disponible en los generadores térmicos convencionales acoplados en el sistema.

Además de la variabilidad, otra particularidad de la energía eólica es la relativa facilidad para desconectarse ante perturbaciones en el sistema y, en particular, ante huecos de tensión que aparecen en el sistema eléctrico por cortocircuitos que surgen como consecuencia de faltas en el sistema eléctrico o cualquier tipo de avería que se pueda producir. En el tiempo en el que las protecciones del sistema tardan en despejar esa falta, se produce un hueco de tensión que afecta a los aerogeneradores que no cumplen los requisitos establecidos en el procedimiento de operación 12.3 “Requisitos de respuesta frente a huecos de tensión de las instalaciones eólicas” y se desconectan de la red. El volumen de este grupo de generadores es muy importante en España y puede comprometer seriamente la seguridad del sistema. Todo esto hace que sea precisa una necesidad de supervisión y control de esta producción muy particular.

Además, la energía eólica tiene otra característica que la diferencia de la generación de régimen ordinario y es la gran cantidad de unidades de producción existentes, en comparación con las unidades de generación convencional, considerando la misma potencia instalada. Si nos vamos al caso extremo, podemos pensar, por ejemplo, que 17.000 MW nucleares estarían constituidos por unas 16 ó 17 unidades o generadores. Estos mismos 17.000 MW eólicos suponen del orden de 670 parques eólicos, lo que conlleva una mayor complejidad en la gestión de tal cantidad de unidades independientes, con múltiples gestores y sobre todo concentradas en ciertos lugares del territorio español. Por otra parte, esta situación provoca que surjan problemas de evacuación y transporte de la energía generada por ellos, ya que el desarrollo de la red eléctrica no cuenta con el apoyo de todas las administraciones y cuenta con una oposición bastante fuerte, por parte de los grupos ecologistas.

Esta distribución desigual de la generación eólica en el territorio español, concentrándose gran parte de ella en 3 ó 4 zonas concretas, hace que en momentos de alta producción eólica las

líneas de transporte existentes sean insuficientes para evacuar toda la generación y se produzcan sobrecargas en ellas, en tiempo real y ante el fallo de ciertas líneas. [7]

Este proyecto puede ayudar a prever dichas sobrecargas en las líneas; para ello será necesario analizar la viabilidad del método propuesto.

Aquí se muestra un mapa de España en el que se puede ver la producción por zonas y se observa lo dicho anteriormente, la distribución desigual de la generación eólica en España, figura 1. [8]

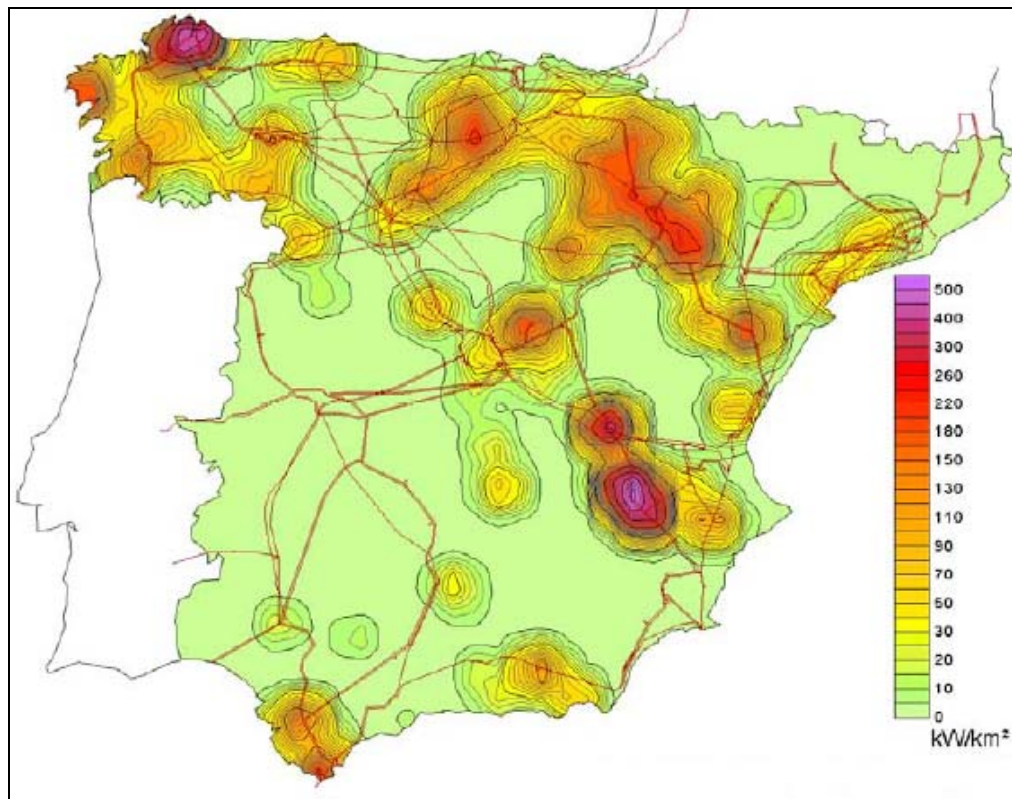


Figura 1. Distribución de la generación eólica en España.

Evolución de la potencia instalada en España en los últimos 12 años, figura 2. [8]

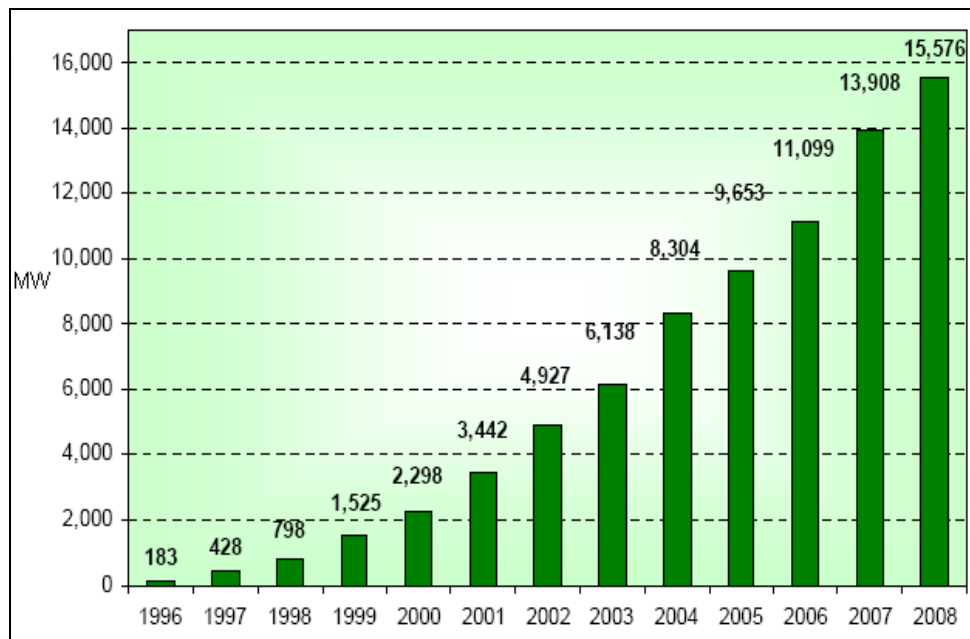


Figura 2. Evolución de la potencia eólica instalada.

Otras características del sistema eléctrico español, que hasta ahora no se han mencionado son:

- Demandas medias, en torno a los 34.000 MW, con puntas de 44.000 MW y valles de 18.000 MW.
- De los 17.000 MW de generación eólica instalados, el valor máximo de producción eólica alcanzado está en torno a los 11.000 MW, dándose cada vez con más frecuencia, 5 ó 6 veces al año, valores de producción de 9.000 – 10.000 MW.

Estos valores elevados de generación eólica se dan, en ocasiones, durante la noche o fin de semana, cuando la demanda de energía es más baja y por lo tanto la potencia aportada por la generación convencional es algo más del 50 %. Esto implica que la capacidad de regulación del sistema ante un incidente que derive en la pérdida de energía eólica, es muy baja; con lo que el riesgo para el sistema aumenta.

Por tanto otro de los principales problemas que se ha de abordar es el de integrar la máxima energía eólica en el sistema eléctrico español, de forma compatible con la seguridad de dicho sistema.

Para ello se han desarrollado herramientas que permiten a los gestores de los parques eólicos, de la red distribución y de transporte contar con previsiones de generación eólica bastante acertadas y programas de cálculo de pérdida de generación eólica ante perturbaciones en la red, entre otras. Pero todavía es necesario seguir trabajando en nuevas herramientas que permitan obtener predicciones de generación más precisas, de forma que se pueda considerar la generación eólica como la convencional, a la hora de gestionar su programación y su producción futura y que así, no introduzca incertidumbre en la programación y riesgos en el sistema eléctrico.

Las herramientas actuales de previsión eólica con que cuentan los gestores de los parques eólicos y de la red de transporte son sistemas basados en las previsiones meteorológicas, los cuales mediante pronósticos facilitan las previsiones de generación eólica por parques, zonas geográficas y para el global de la generación eólica instalada en España. [7]

Con estos resultados de producción eólica son con los que los gestores de los parques eólicos acuden al mercado diario e intradiarios, a vender su energía.

Funcionamiento de los mercados de energía eléctrica y operación en tiempo real del sistema eléctrico.

El mercado de energía eléctrica en España está basado en un mercado diario, 6 mercados intradiarios, un mercado de secundaria, 6 mercados de desvíos y 24 mercados de terciaria. Al mercado diario y a los 6 intradiarios acuden los productores de energía para venderla y los consumidores, en este caso las distribuidoras y comercializadoras a comprar la energía que necesitan para abastecer a sus clientes.

El mercado diario tiene lugar el día anterior (D-1) al día (D) para el que se está vendiendo y comprando energía; tiene como horizonte las 24 horas del día siguiente (D) y se cierra a las 10:30 h del día (D-1). A este mercado acuden, entre otros, los gestores de los parques eólicos a vender la energía que según sus previsiones se producirá el día siguiente, día (D). En el mercado diario es en el que se mueve la mayor cantidad de energía.

Una vez cerrado este mercado, el OMEL (Operador del Mercado Eléctrico) publica antes de las 12:00 h el resultado en el PBF (Programa Base de Funcionamiento). Este programa lo forman el programa casado para todos los generadores del sistema y los intercambios internacionales.

Con el PBF Red Eléctrica realiza un estudio de restricciones para evaluar los posibles problemas que pueden surgir en la red. Si aparecen problemas de congestión, sobrecargas, etc... se modifica la generación convencional y los programas de intercambio internacionales para eliminar las restricciones técnicas y cumplir los criterios de seguridad en la operación del sistema. Si los problemas son causados por la generación eólica, se tendrán en cuenta y en tiempo real se actuará si es necesario, ya que debido a la incertidumbre de las previsiones eólicas, puede que en tiempo real la generación eólica sea menor que la considerada, en el momento del estudio de restricciones.

Una vez realizado el estudio de restricciones y realizadas las modificaciones necesarias sobre el PBF para cumplir los criterios de seguridad, antes de las 14:00 h, Red Eléctrica publica el PVP (Programa Viable Provisional).

A las 15:30 h se cierra el mercado de secundaria, al que los generadores convencionales: térmicos e hidráulicos, acuden a presentar sus ofertas de regulación secundaria a subir y a bajar. Antes de las 16:00 h Red Eléctrica publica el resultado de dicho mercado, que unido al PVP conforma el PVD (Programa Viable Definitivo).

El mismo día (D-1) existen dos mercados intradiarios y el día (D) cuatro. Éstos mercados funcionan como el mercado diario y tienen la función de ajuste de la energía que se ha negociado en los anteriores mercados. Son utilizados por los gestores de los parques eólicos para modificar la energía eólica vendida, ya que las previsiones de generación pueden cambiar y son más fiables a medida que se acerca el horizonte temporal para el cual se calculan dichas previsiones. Una vez cerrados cada uno de estos mercados Red Eléctrica realiza un análisis de restricciones técnicas y modifica si es necesario la generación convencional y los programas internacionales. Con los programas resultantes del estudio de restricciones Red Eléctrica publica el PHF (Programa Horario Final).

Después de esto si la generación casada en el PHF correspondiente, es insuficiente o muy superior a la demanda prevista, Red Eléctrica convoca un mercado de desvíos, cuyo horizonte de aplicación es hasta el siguiente mercado intradiario. A este mercado de desvíos, los generadores convencionales acuden a presentar sus ofertas. Una vez cerrado el mercado de desvíos se publica la asignación de las ofertas.

15 minutos antes de cada hora, Red Eléctrica con los últimos datos de la previsión eólica, solar y de la demanda realiza un ajuste fino de la generación a la demanda prevista para la siguiente hora.

La generación eólica que se vendió en el último mercado intradiario, de acuerdo a las previsiones que obtuvieron los gestores de los parques, puede que no se aproxime a la generación eólica que se está produciendo en ese momento; por lo que el operador del sistema debe aumentar o disminuir la generación convencional para compensar esa diferencia de generación.

Todo esto lo realiza con el mercado de terciaria al que acuden los generadores convencionales. Este mercado se cierra en el minuto 35 de la hora anterior y antes del minuto 45 se publica el PHO (Programa Horario Operativo). En el figura el programa definitivo que deberán cumplir las unidades de generación y los programas de intercambio internacionales.

Dentro de cada hora puede que sea necesario realizar nuevos ajustes de la generación a la demanda, para ello se utilizarán las ofertas del último mercado de terciaria y se publica un nuevo PHO.

En tiempo real pueden aparecer restricciones técnicas que se solucionan modificando en ese momento, por seguridad, la generación o programas internacionales.

El desajuste entre la previsión de generación eólica y la producción real provoca, como ya se ha mencionado, el consumo de recursos del sistema e introduce incertidumbre y riesgos en el sistema. Es por eso que se debe seguir trabajando en nuevas herramientas de previsión eólica.

Aquí se muestran dos gráficos en los que se pueden ver el funcionamiento de los mercados de producción y los periodos de ejecución junto con los horizontes de los mercados. [8]



Figura 3. Mercados de energía eléctrica

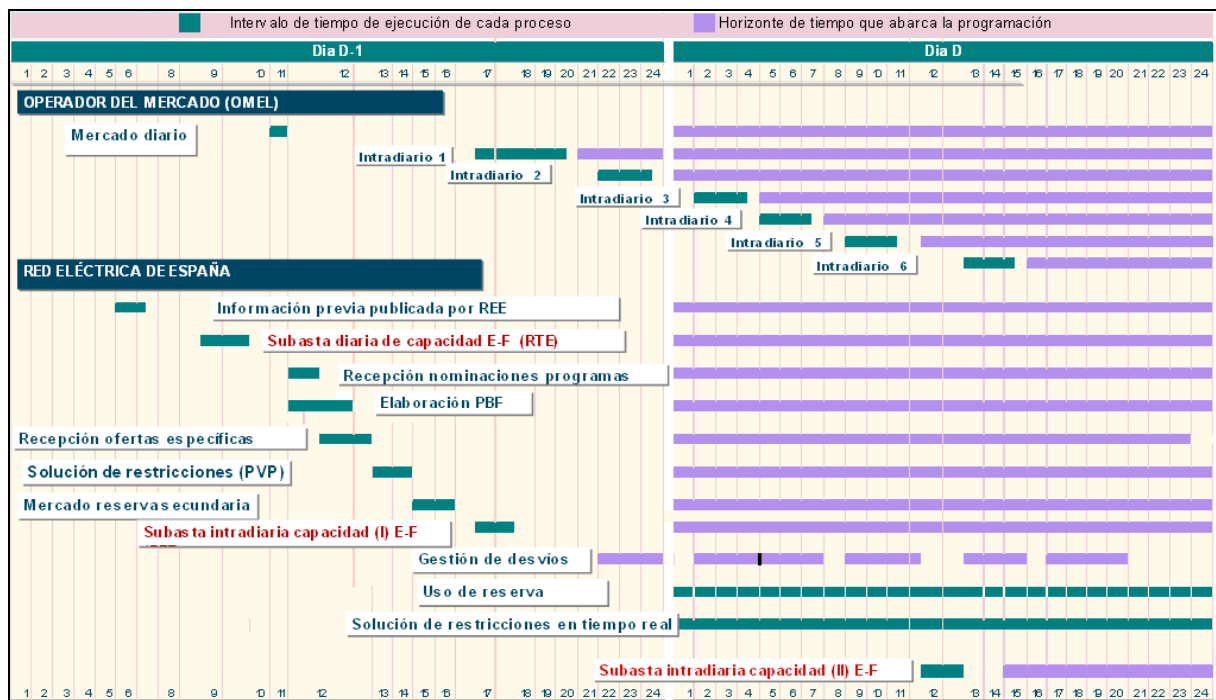


Figura 4. Periodos de ejecución y horizontes de los mercados de energía.

3. Análisis de la incertidumbre de las previsiones de generación eólica.

Actualmente Red Eléctrica cuenta con un programa de previsión de generación eólica (SIPREÓLICO) [11], desarrollado en colaboración con la Universidad Carlos III de Madrid, que le ayuda a la hora de realizar los análisis de restricciones y las programaciones horarias. SIPREÓLICO es una herramienta de previsión eólica para un horizonte temporal de 48 horas. Se alimenta de las previsiones meteorológicas (velocidad y dirección del viento) del Instituto Nacional de Meteorología y del Centro Europeo y de datos históricos de producción de energía eólica. Utilizando métodos estadísticos (series de tiempo) y las curvas de potencia activa de los aerogeneradores, proporciona una previsión: por parques, zonas geográficas y para el global de la generación eólica instalada en España, para cada hora, con un horizonte de previsión de 48 horas.

En la siguiente gráfica, figura 5, se observa como el error en la previsión dada por SIPREÓLICO, aumenta al aumentar el horizonte temporal para el cual se realiza dicha previsión.

Esta gráfica se ha obtenido al comparar, desde junio hasta diciembre de 2005, las previsiones obtenidas con SIPREÓLICO, con los valores de producción eólica real [11].

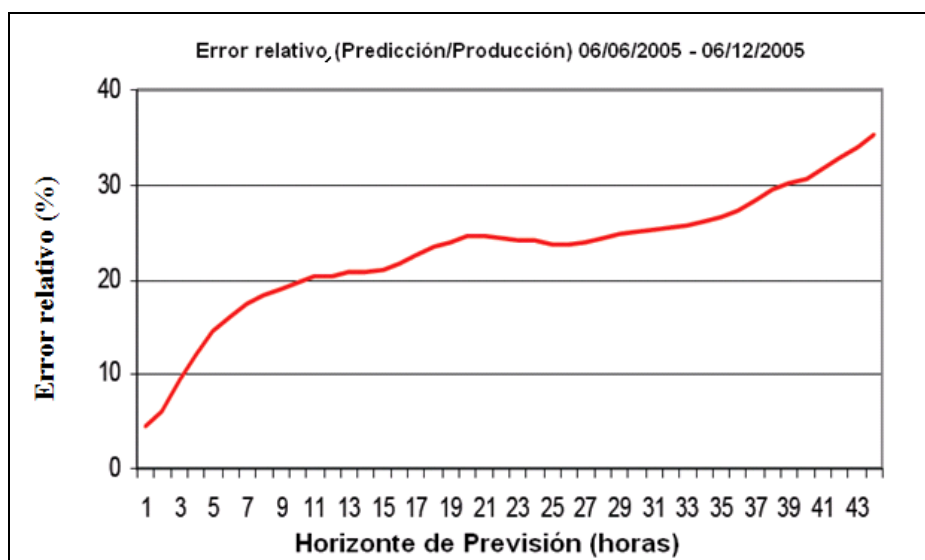


Figura 5. Error relativo en las previsiones obtenidas con SIPREÓLICO.

En general las previsiones hechas por cualquier programa de previsión son inciertas y esta incertidumbre aumenta cuanto mayor es el horizonte temporal de dicha previsión.

Los programas de previsión trabajan con un horizonte temporal de 48 - 72 horas y conviene saber cuál es la incertidumbre de las previsiones a la hora de utilizarlas en estudios de seguridad y restricciones.

Se han realizado estudios con datos de previsiones y producciones reales para esas previsiones, para calcular la incertidumbre de dichas previsiones, según el horizonte temporal de éstas; ya que de forma analítica es complicado calcular la incertidumbre de las previsiones. Estos estudios han demostrado que la incertidumbre de las previsiones para un horizonte temporal pequeño es menor, que para un horizonte temporal mayor y dicha incertidumbre es independiente de la potencia generada prevista, para un horizonte temporal concreto.

Un aspecto a tener en cuenta es la correlación entre las incertidumbres de las previsiones de generación de parques eólicos cercanos geográficamente. Ya que al ser las previsiones meteorológicas similares para ellos, las potencias previstas estarán relacionadas. Todavía no existen estudios que calculen la dependencia entre estas incertidumbres pero lo que sí se ha comprobado, con datos de producciones de parques eólicos cercanos, es la dependencia de sus potencias generadas. Para este proyecto será necesario tener en cuenta estas dependencias entre parques cercanos, para conseguir una mayor aproximación al comportamiento real.

Por el momento conseguir una certeza total en las previsiones parece difícil, por lo que dicha incertidumbre se asume como una buena aproximación. [1]

4. Descripción y nociones básicas del programa PSS/E.

En este apartado se va a describir la estructura y funcionamiento del programa de simulación de sistemas eléctricos, PSS/E. [4]

Para la realización de flujos de cargas es necesario partir de un modelo de red eléctrica.

En PSS/E el modelo de red se introduce cargando un caso, es decir, abriendo un archivo, que contiene dicho modelo de red, para que el programa lo lea.

En este caso se cargará un caso con el PBF (programa base de funcionamiento).

El modelo de red, para PSS/E, tiene la siguiente estructura:

La red eléctrica se puede dividir en áreas, de manera que cada área corresponda a una zona geográfica concreta y en ellas se encuentren ubicadas las diferentes subestaciones, cargas, generadores, etc... En la red eléctrica de alta tensión española se han definido 5 zonas, y 3 zonas adicionales para los modelos de red eléctrica de Francia, Portugal y Marruecos.

A cada área se la asigna un número:

<u>Número de área</u>	<u>Nombre del área</u>
1	NOROESTE
2	NORTE
3	ESTE
4	CENTRO
5	SUR
6	PORTUGAL
7	ONE
8	FRANCIA

También se pueden definir zonas, que en el caso de la red española corresponden a las comunidades autónomas, salvo excepciones. Como en el caso de las áreas, a cada zona se le asigna un número.

Esta partición del modelo de red en áreas y zonas resulta útil a la hora de estudiar problemas concretos en un parte determinada de la red, ya que permite modificar parámetros, correr contingencias y mostrar resultados por áreas o zonas.

En PSS/E cada subestación, carga y generador es identificado con un número; se conoce como “número de bus”, ya que en PSS/E cada elemento es considerado un nudo, el cual mediante líneas se conecta a otros nudos. A un subestación llegan líneas procedentes de otra subestación, de cargas y de generadores, por lo tanto al introducir el número de bus de una subestación aparecerán los números de bus de los elementos que a ella están conectados.

En el sistema español se ha seguido un criterio para numerar. El primer número del número de bus corresponde, en el caso de las subestaciones, al nivel de tensión y el segundo número al área al cual pertenece la subestación:

- 1 para 400 kV
- 2 para 220 kV.

De esta forma resulta más fácil identificar los números de bus y a que zona pertenece el elemento al que representan.

Las líneas y los transformadores se representan mediante los dos números de bus de las subestaciones que unen.

Existe una forma de identificar unívocamente a las cargas, generadores, transformadores y líneas que es el “identificador”. En una misma subestación pueden existir varias cargas, generadores o transformadores con el mismo número de bus y varias líneas que unan dos subestaciones (dobles o triples circuitos); ya que el elevado número de estos elementos en un sistema eléctrico como el español, hace imposible identificarlos coherentemente sólo con el número de bus, se les asigna por tanto, el mismo número de bus pero un identificador diferente: 1, 2, 3, 4, 5 etc...

A la hora de trabajar con estos elementos, para distinguirlos, basta elegir el número de bus deseado y el identificador que le corresponda.

De cada elemento PSS/E almacena y trabaja con sus valores característicos ya sea una subestación: voltaje, ángulo, nivel de tensión, área y zona, o un generador: estado de conexión, potencia activa reactiva, potencias máxima y mínimas, valores eléctricos de resistencia y reactancia etc...

Todos estos valores pueden visualizarse a petición del usuario y ser utilizados en programación como es el caso de este proyecto.

La interfaz de PSS/E versión 31 es muy parecida a cualquier programa de office. En la figura 6 se muestra la pantalla que aparece al arrancar el programa.

Bus Name	Id	Code	In Service	Pgen (MW)	Pmax (MW)	Pmin (MW)	Qgen (Mvar)	Qmax (Mvar)	Qmin (Mvar)	Mbase (MVA)	R Source (pu)	X Source (pu)	RTran (pu)	XTr (pu)
VALPAR52	15.000	1	4	0.0000	32.0000	2.0000	0.0000	12.0000	-7.0000	33.20	0.000000	0.190000	0.000000	0.0
VALPAR53	15.000	1	4	0.0000	33.0000	2.0000	0.0000	12.0000	-7.0000	33.20	0.000000	0.190000	0.000000	0.0
VILLAR54	13.800	1	4	0.0000	31.0000	9.0000	0.0000	12.0000	-8.0000	32.00	0.000000	0.220000	0.000000	0.0
VILLAR53	13.800	1	4	0.0000	32.0000	9.0000	0.0000	12.0000	-8.0000	32.00	0.000000	0.220000	0.000000	0.0
VILLAR6	13.800	1	2	25.5900	32.0000	9.0000	0.7127	12.0000	-8.0000	32.00	0.000000	0.220000	0.000000	0.0
VILLAR52	15.000	1	4	0.0000	110.0000	30.0000	0.0000	45.0000	-30.0000	125.00	0.000000	0.250000	0.000000	0.0
VILLAR54	13.800	1	4	0.0000	139.0000	90.0000	0.0000	45.0000	-25.0000	169.00	0.000000	0.245000	0.000000	0.0
VILLAR9	13.800	1	2	-137.2300	135.0000	90.0000	-1.3432	45.0000	-25.0000	169.00	0.000000	0.245000	0.000000	0.0
VILLAR53	13.800	1	4	0.0000	135.0000	90.0000	0.0000	45.0000	-25.0000	169.00	0.000000	0.245000	0.000000	0.0
VILLAR52	13.800	1	4	0.0000	135.0000	90.0000	0.0000	45.0000	-25.0000	169.00	0.000000	0.245000	0.000000	0.0
VILLAR6	13.800	1	2	-141.2500	145.0000	100.0000	4.7553	45.0000	-25.0000	169.00	0.000000	0.245000	0.000000	0.0
VILLAR51	13.800	1	4	0.0000	149.0000	100.0000	0.0000	45.0000	-25.0000	169.00	0.000000	0.245000	0.000000	0.0
ZABALG52	12.600	1	4	0.0000	50.0000	0.0000	0.0000	25.0000	-15.0000	55.90	0.000000	0.140000	0.000000	0.0
ZABALG53	12.600	1	4	0.0000	50.0000	0.0000	0.0000	25.0000	-15.0000	55.90	0.000000	0.140000	0.000000	0.0
ZERBE4	15.750	1	2	198.2800	271.3000	195.0000	29.0457	180.0000	-120.0000	354.20	0.000000	0.245100	0.000000	0.0
ZERBE3	15.750	1	2	124.3800	257.4000	125.0000	19.2483	160.0000	-110.0000	318.20	0.000000	0.223100	0.000000	0.0
ZERBE2	15.750	1	2	122.6900	256.6000	125.0000	46.0352	160.0000	-110.0000	318.20	0.000000	0.223100	0.000000	0.0
PERELL1	110.00	1	2	20.1100	28.0000	0.0000	-4.4052	10.0000	-10.0000	30.00	0.000000	0.168000	0.000000	0.0
PERELL1	110.00	2	2	20.1100	28.0000	0.0000	-4.4052	10.0000	-10.0000	30.00	0.000000	0.168000	0.000000	0.0
ARTES1	110.00	5	2	13.3800	20.0000	5.0000	-1.7089	24.0000	-24.0000	8.50	0.000000	0.200000	0.000000	0.0
ARTES1	110.00	6	2	13.4800	20.0000	5.0000	-1.7216	24.0000	-24.0000	33.50	0.000000	0.200000	0.000000	0.0
ARTES1	110.00	7	2	9.4800	11.0000	0.0000	-1.2108	5.0000	-5.0000	8.50	0.000000	0.200000	0.000000	0.0

Figura 6. Pantalla de inicio de PSS/E

En ella se encuentra una barra superior desplegable y numerosos botones con las acciones más utilizadas. En la parte central se pueden ver los parámetros de subestaciones, cargas generadores, líneas, etc... Es en esta pantalla donde el usuario puede modificar los valores de los parámetros si lo desea y en la parte inferior aparece la pantalla de proceso en la que se puede visualizar la ejecución de los procesos y resultados. [4]

Para ejecutar los programas creados en este proyecto y obtener los resultados de los flujos de cargas probabilístico, será necesario abrir una nueva pantalla desde esta inicial, con el botón “CLI” Command Line Input. Esta nueva pantalla, figura 7, nos permite escribir instrucciones para el programa o ejecutar programas ya creados en lenguaje PSS/E o Python. Bastará con pulsar el botón “CLI” y se abrirá la siguiente pantalla:

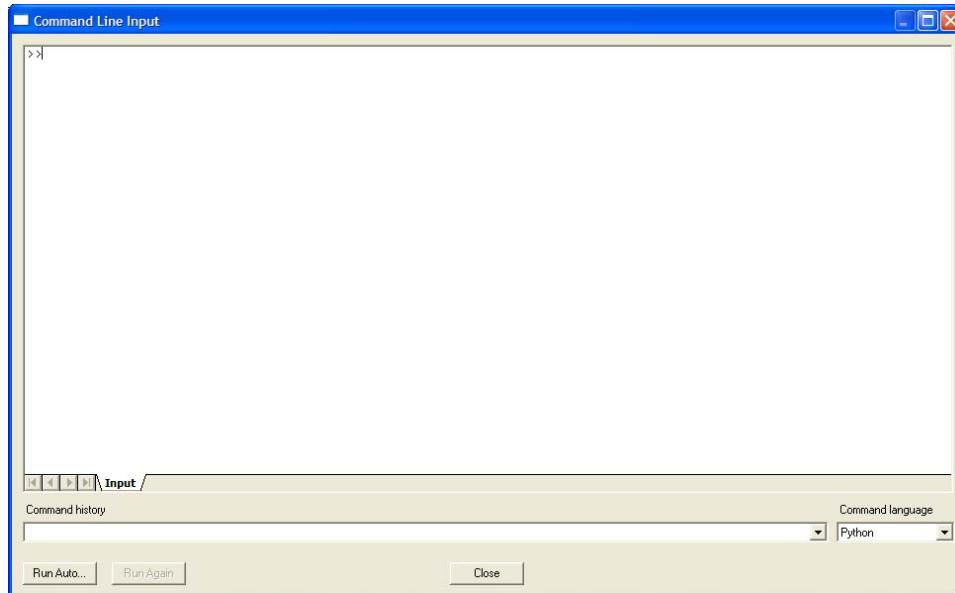


Figura 7. Pantalla de entrada de comandos

Con el botón “Run Auto” se elige el programa y al seleccionarlo este se ejecuta. Los resultados dependiendo del programa ejecutado aparecerán en la pantalla inicial, en un archivo de texto o impresos, entre otros. [4]

5. Descripción del flujo de cargas probabilístico.

El flujo de cargas probabilístico es una de las herramientas probabilísticas más conocidas para simular el comportamiento futuro de las variables de un sistema eléctrico: tensión en las subestaciones y potencia por las líneas.

Existen varios métodos para resolverlo; a continuación se exponen algunos de ellos.

La forma más sencilla de resolver un flujo de cargas probabilístico es mediante el método de Monte Carlo. Esta técnica implica repetidas simulaciones utilizando los valores de las variables del sistema, obtenidos a partir de la FDP de cada variable considerada.

El inconveniente de este método es que son necesarias muchas simulaciones, para conseguir una representación adecuada del sistema real que se esté estudiando. Lo que lo hace poco práctico por el tiempo que se necesita para obtener los resultados.

Un método alternativo para resolverlo es la convolución de la FDP de las variables aleatorias consideradas, cuando éstas son independientes unas de otras y linealmente relacionadas. Aunque esto reduce la carga computacional, resulta difícil obtener la FDP de una sola línea cuando se consideran varias inyecciones de potencia aleatorias en el sistema.

Puede resolverse utilizando las técnicas de la transformada rápida de Fourier (FFT) que se propusieron para reducir la carga computacional, pero este método está vinculado a la técnica de convolución y no resuelve el problema de manera eficiente. Ciertos estudios consideran la incertidumbre de la carga y proponen el uso de la FFT y la convolución en las redes de distribución, haciendo una simplificación de la estimación de la FDP para las previsiones a corto plazo de la generación.

Una propuesta reciente es el método de estimación de punto que aproxima los momentos de las variables de interés.

Todos estos enfoques consideran independientes a las variables aleatorias. Sin embargo, debe considerarse la dependencia entre las incertidumbres de la inyección de potencia eólica. La generalización de algunos de estos métodos para el análisis de la dependencia entre las variables aleatorias es muy compleja o inviable. Hay algunas propuestas que consideran que esta dependencia, sólo existe entre las cargas; siguiendo una relación lineal o teniendo en cuenta la covarianza en las ecuaciones.

La incertidumbre de la carga se incluye en la mayoría de los métodos para resolver el flujo de cargas probabilístico. Esta incertidumbre normalmente no es muy alta, especialmente para las previsiones del día siguiente y puede ser modelada utilizando la función de densidad de probabilidad gaussiana.

La proliferación de la energía eólica, sin embargo, plantea nuevos retos, ya que la variabilidad en la producción de esta energía es mucho mayor que la de la carga. Por lo general la FDP de la variable potencia eólica, no es de gaussiana, suele modelarse como una FDP beta y las potencias eólicas generadas por los parques eólicos están muy correlacionadas; ya que los parques de una misma zona geográfica se ven afectados por condiciones meteorológicas similares.

Los estudios de planificación a largo plazo deben considerar la FDP basada en las distribuciones de Weibull, mientras que el análisis de la operación a corto plazo necesita utilizar una FDP cuya estimación está todavía en estudio.

El uso de cumulantes y la aproximación de una FDP por series ortogonales (series de expansión Gram-Charlier A) también se han propuesto recientemente para resolver un flujo de cargas probabilístico. Tiene propiedades interesantes y es computacionalmente eficiente. Para las grandes redes de transmisión de energía, parece que este enfoque es muy adecuado debido a los bajos requisitos computacionales. Tiene la desventaja de la linealización necesaria, pero puede ser fácilmente generalizado para variables aleatorias dependientes. Sin embargo, para FDP no gaussianas, las series de expansión de Gram-Charlier A, tienen graves problemas de convergencia. Otros enfoques como la expansión Cornish Fisher, dan mejores resultados sin más carga computacional.

Un estudio reciente propone un método analítico, para resolver el flujo de cargas probabilístico, llamado método lineal reforzado. Este método puede aplicarse a los sistemas eléctricos donde se considera la incertidumbre de la potencia eólica, de la carga y la disponibilidad de la generación convencional. Matemáticamente, esto significa que las variables aleatorias consideradas pueden ser continuas y/o discretas y también dependientes y/o independientes. [1]

Para la realización de este proyecto se ha tomado como sistema de estudio el sistema eléctrico español y en él se tratará de analizar la viabilidad y eficacia de la aplicación del flujo de cargas probabilístico mediante el método de Monte Carlo [2], en los estudios de restricciones y tiempo real que el operador del sistema realiza. De manera que a partir de las previsiones de generación eólica, dadas por los gestores de los parques o por el programa de previsión del operador del sistema (SIPREÓLICO), se pueda obtener la probabilidad de sobrecarga en las líneas de transporte y la probabilidad de estar fuera de un cierto rango de tensiones en las diferentes subestaciones de la red de transporte.

Se ha elegido resolver los flujos de cargas probabilísticos mediante el método de Monte Carlo [2] por su simplicidad de cálculo, ya que no requiere de procesos de cálculo complejos ni del manejo de complicados algoritmos matemáticos y por lo tanto la carga computacional de cada simulación es relativamente baja.

Se pretenden conocer los tiempos de cálculo necesarios mediante este método, de manera que se pueda analizar la viabilidad de su aplicación; ya que como se ha dicho anteriormente el método de Monte Carlo necesita realizar numerosas simulaciones y esto hace que los tiempos de cálculo sean más altos que los necesitados por otros métodos.

5.1. El método de Monte Carlo.

5.1.1. Principios básicos.

El Método de Monte Carlo permite resolver problemas físicos y matemáticos mediante la simulación de variables aleatorias y es una herramienta habitual en el análisis de sistemas multidimensionales complejos. Puede ser utilizado para resolver tanto problemas estocásticos como deterministas, aunque la aplicación a los primeros es más usual. Este método está basado en un proceso iterativo, por lo que en cada paso se utiliza un nuevo conjunto de valores de las variables aleatorias involucradas. Estos valores son generados de acuerdo a la FDP de cada variable.

Seguidamente se presentan los fundamentos del método y algunos conceptos y definiciones relacionadas con su aplicación.

- Una aproximación estadística comprende dos valores, el error y el margen de confianza de dicho error. El intervalo de confianza para el estimador de una variable aleatoria es el intervalo definido por el error entorno a dicho estimador. La probabilidad de que el valor correcto del estimador se encuentre en ese intervalo es el nivel de confianza.
- La media de una muestra, \bar{h}_n , de una variable aleatoria $h(x)$ converge en probabilidad hacia el valor medio de la variable.
- La utilización de un procedimiento de muestreo adecuado para la obtención de una variable aleatoria no presenta diferencias respecto a la generación de la misma variable mediante un proceso controlado por la propia FDP de la variable.
- El error estadístico de una solución obtenida mediante el método de Monte Carlo converge hacia cero a un ritmo proporcional a $(\sqrt{n})^{-1}$, y la FDP de la variable converge hacia una distribución normal, tal como se ha mencionado más arriba. Esto significa que la distribución del valor medio de una muestra de un estimador es independiente de la distribución del propio estimador.
- El método de Monte Carlo converge más lentamente que un método numérico, por ejemplo para el cálculo de integrales, no obstante, la convergencia es independiente de la dimensión del sistema.

El objetivo del método de Monte Carlo es obtener la respuesta de un sistema en función de algunas variables aleatorias de entrada. El proceso de muestreo es repetido de forma iterativa hasta alcanzar la convergencia. Si la generación de las variables de entrada se lleva a cabo mediante un procedimiento adecuado, la solución converge a medida que el número de muestras aumenta, $n \rightarrow \infty$, siendo $(\sqrt{n})^{-1}$ la velocidad con la que el error estadístico converge hacia cero. [2]

5.1.2. El teorema central del límite.

Como se ha dicho anteriormente el método de Monte Carlo está basado en un proceso iterativo, por tanto ha de calcularse el número de iteraciones mínimo para que los resultados sean fiables y estén dentro del error admitido.

Si para resolver cualquier problema mediante este método se ejecutasen un número de iteraciones infinito, los resultados tendrían una fiabilidad del 100 % (un margen de error cero) ya que se habrían considerado todos los valores posibles de las variables, pero esto resulta imposible de realizar ya que los tiempos de cálculo y la carga computacional serían enormes.

Para obtener resultados dentro de un margen de error admisible basta con realizar los cálculos un número de iteraciones mínimo.

Para justificar esto se puede utilizar el teorema central del límite, que se enuncia de la siguiente forma:

Sea X (la carga de una línea de transporte eléctrico) una variable aleatoria. Todos los valores que pueda tomar dicha variable forman una población con media μ y desviación típica σ , de la que se extraen aleatoriamente todas las posibles muestras, todas ellas de tamaño n (número de iteraciones). Obteniendo las medias de todas estas muestras y considerándolas una distribución de datos (la distribución muestral de medias), se comprueba que:

a) La media de los datos es la media μ de la población, es decir la media de las medias de las muestras, es igual que la media de la población.

b) Estas medias se distribuyen alrededor de la media de la población, con una desviación típica, llamada *desviación típica de la media*, igual a la de la población dividida por la raíz de n , es decir, la distribución típica de la media es: σ/\sqrt{n}

c) La distribución de las medias muestrales, es una distribución de tipo "normal", siempre que la población de procedencia lo sea, o incluso si no lo es, siempre que el tamaño de las muestras sea mayor o igual a 30.

En consecuencia, si una población tiene media μ y distribución típica σ y tomamos muestras de tamaño n (de tamaño al menos 30 o cualquier tamaño, si la población sigue una distribución "normal"), las medias de estas muestras siguen aproximadamente la distribución:

$$N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$$

Además, cuanto mayor es el valor de N , mejor es la aproximación "normal".

Se ha nombrado un concepto importante: la distribución típica de la media σ/\sqrt{n} , que es el grado de variabilidad de las medias muestrales. Cuanto menor sea, más ajustadas a la media de la población serán las medias que obtengamos de las muestras. De su propia definición, es fácil darse cuenta de que cuanto mayor es el tamaño de la muestra, menor es este grado de

variabilidad y por tanto más similar a la media de la población será la media obtenida de la muestra.

Por tanto si la medida de interés es la carga de una línea de transporte, es posible estimar el número de iteraciones necesario para garantizar un margen de error determinado en dicha medida y para encontrar el valor medio de la carga de la línea, con un cierto nivel de confianza y para un margen de error determinado.

Para ello se utilizará la siguiente expresión:

$$N = \frac{Z_{\alpha/2}^2 \cdot p(1-p)}{s^2}$$

donde:

- p es la probabilidad de que la línea de transporte supere un nivel de carga determinado.
- s es el error admisible o la precisión de que los resultados obtenidos se encuentren dentro del intervalo establecido, en %
- $Z_{\alpha/2}$ es el valor que en una distribución normal de valor medio nulo y desviación estándar unidad define un intervalo $[-Z_{\alpha/2}, +Z_{\alpha/2}]$ dentro del cual se encuentra la variable aleatoria con un probabilidad $(1 - \alpha)$.
- N es el número de iteraciones que garantiza un nivel de confianza $(1 - \alpha)$.

5.1.3. *Análisis de convergencia del método.*

Para garantizar que los resultados obtenidos son fiables será necesario, como se ha dicho, estimar el número de iteraciones mínimo a partir del cual el error en las distribuciones de las variables aleatorias, se mantiene por debajo de un valor determinado con cierto margen de confianza. El número hallado tan sólo garantiza que la distribución obtenida para las variables aleatorias de entrada está dentro del margen de error exigido con respecto a la distribución teórica. Es importante destacar que el número de iteraciones necesarias para garantizar el error seleccionado es, a su vez, una variable aleatoria cuya función de distribución presenta una forma lognormal. La figura 8 muestra la función de distribución de una variable lognormal y la función de distribución normal que corresponde al logaritmo de dicha variable. A partir de esta propiedad es posible determinar el número de iteraciones necesarias para alcanzar un nivel de confianza a partir del error máximo especificado. [2]

Supóngase que se desea determinar el número de iteraciones que es necesario realizar para obtener un nivel de confianza del 90% en una variable aleatoria cuya función de densidad acumulada es conocida. El error se acotará fijando 10 puntos en esta función, tal como indica la figura 9. Es decir el error será inferior a un determinado valor cuando la curva de densidad acumulada obtenida se encuentre dentro de la zona acotada por el margen, error aceptado en los 10 puntos escogidos, ver figura 9.

La figura 10 presenta la función de distribución que se obtiene para el número de iteraciones y para su logaritmo neperiano cuando la variable simulada sigue una distribución normal y el

margen de error permitido es del 5% y del 1%, respectivamente. De los resultados obtenidos se deduce que el error en la distribución obtenida es igual o inferior al 1% tras 2300 iteraciones, e igual o inferior al 5% tras 130 iteraciones con un nivel de confianza del 90%. O dicho de otra forma, la probabilidad de que la distribución obtenida tras 2300 iteraciones presente un error inferior al 1% es del 90%. [2]

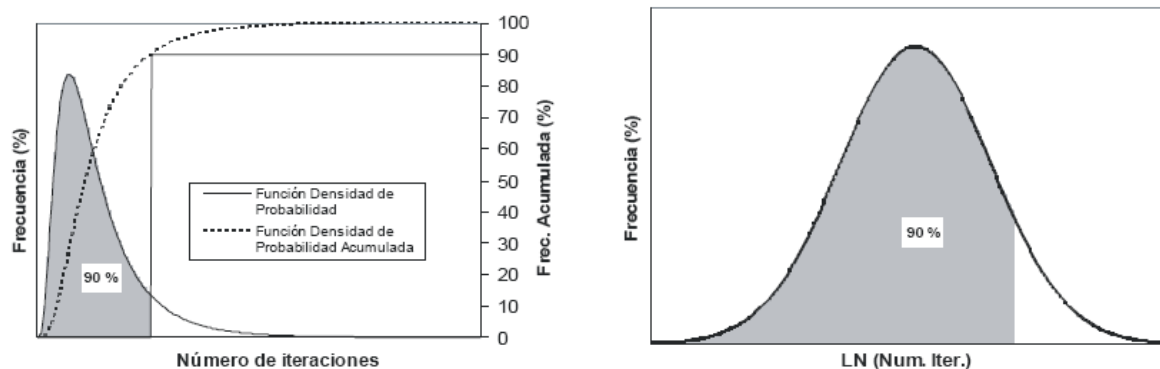


Figura 8. Función de distribución para una variable lognormal.

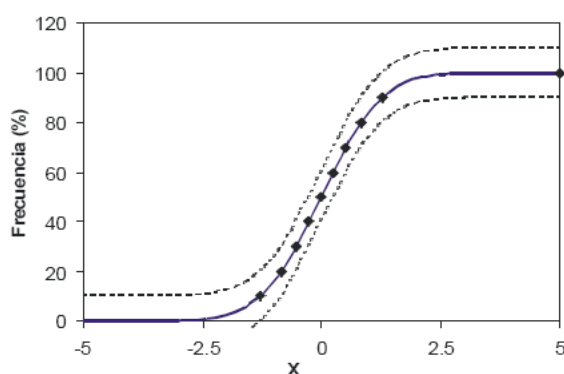


Figura 9. Margen de error permitido en la función de probabilidad acumulada.

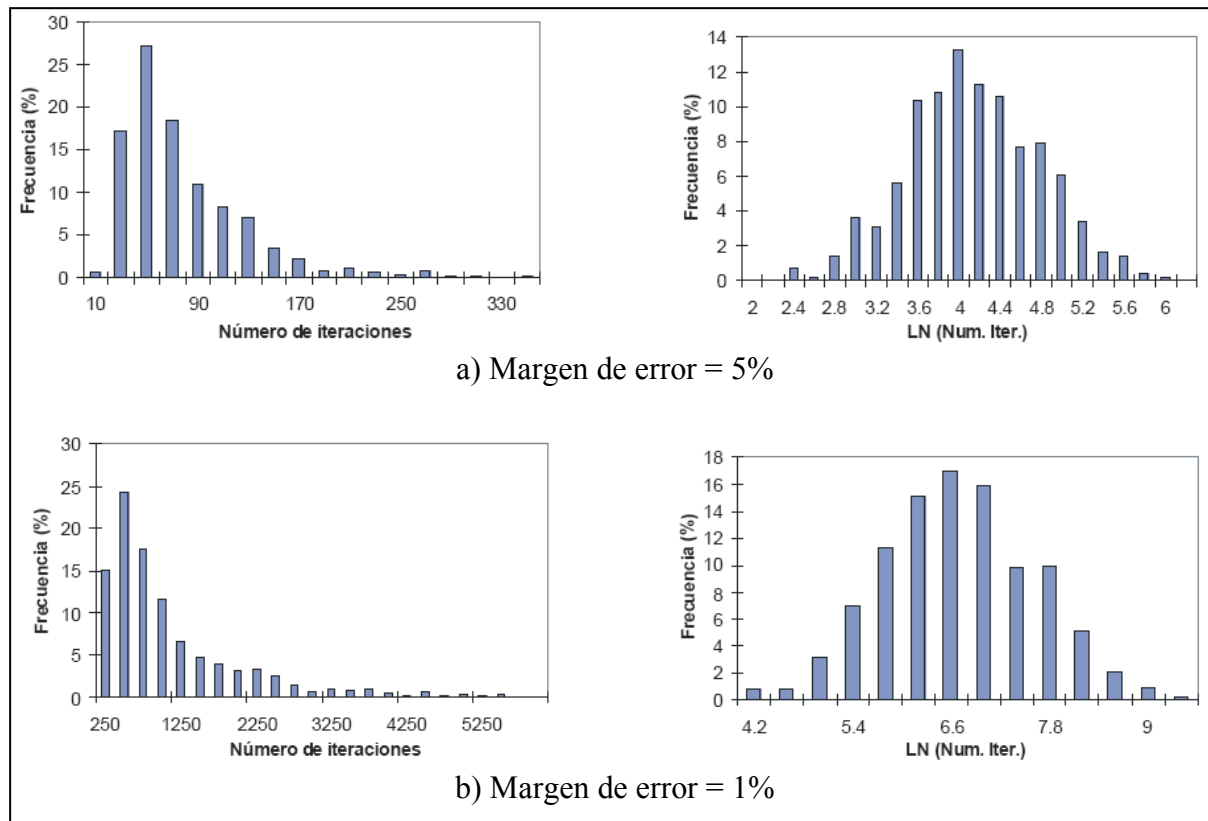


Figura 10. Distribución de iteraciones para un determinado margen de error.

5.2. Modelado del comportamiento de la generación eólica.

Partiendo de la base de que la potencia producida por un generador eólico puede estar entre 0 MW y su potencia nominal, con la misma probabilidad de ocurrencia para cada valor del intervalo; la generación eólica será modelada como una FDP beta, en la que la media de la distribución será la potencia prevista para cada generador eólico.

La expresión analítica de la FDP beta es: [1]

$$f(x; a, b) = \frac{1}{B(a, b)} x^{a-1} (1-x)^{b-1}$$

Donde $B(a, b)$ es la función beta y a, b son parámetros relativos a la media de la distribución, η y la varianza, σ^2 , de manera que: [1]

$$\eta = \frac{a}{a+b} \quad \sigma^2 = \frac{a \cdot b}{(a+b)^2(a+b+1)}$$

La distribución beta para diferentes valores de los parámetros a y b , se representa en la figura 11. [10]

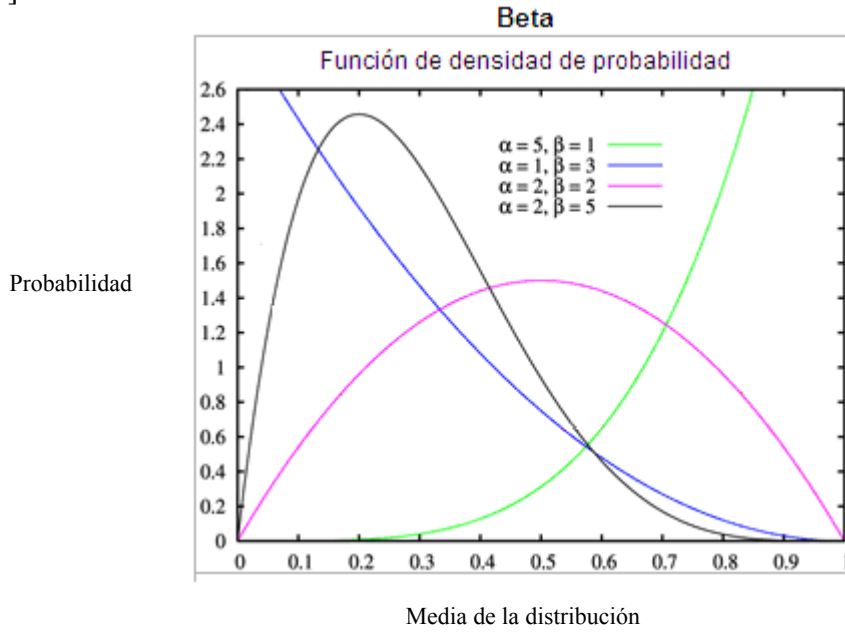


Figura 11. Distribución beta

Debido a que el modelo de red utilizado en este proyecto tiene modelada la generación eólica por parques, no por generadores independientes se tomará cada parque como una unidad independiente y la previsión de generación del parque será la media de la FDP beta.

Esta media, η , será un valor en por unidad (p.u.), que se calculará de la siguiente forma:

$$\frac{P_{prev}}{P_{nom}} = \eta$$

Siendo: P_{prev} = potencia prevista para cada parque eólico, en MW
 P_{nom} = potencia nominal del parque, en MW

La varianza, σ^2 , se calcula a través de la desviación típica, σ , y ésta depende de la potencia generada por el parque eólico, respecto a su potencia nominal. Esta dependencia se ha determinado experimentalmente a partir de los valores de producción históricos de algunos parques eólicos instalados en España. [1]

En la figura 12 se muestra la relación entre la desviación típica, σ , y la potencia generada normalizada a 1. La desviación estándar aparece normalizada a la potencia nominal del parque eólico y la nube de puntos se ha aproximado a la siguiente curva cuadrática: [1]

$$\sigma = -0.79257 \cdot (P_{prev})^2 + 0.77991 \cdot P_{prev} + 0.042078$$

Donde: σ = desviación típica o estándar
 P_{prev} = potencia prevista para cada parque eólico, expresado en p.u.

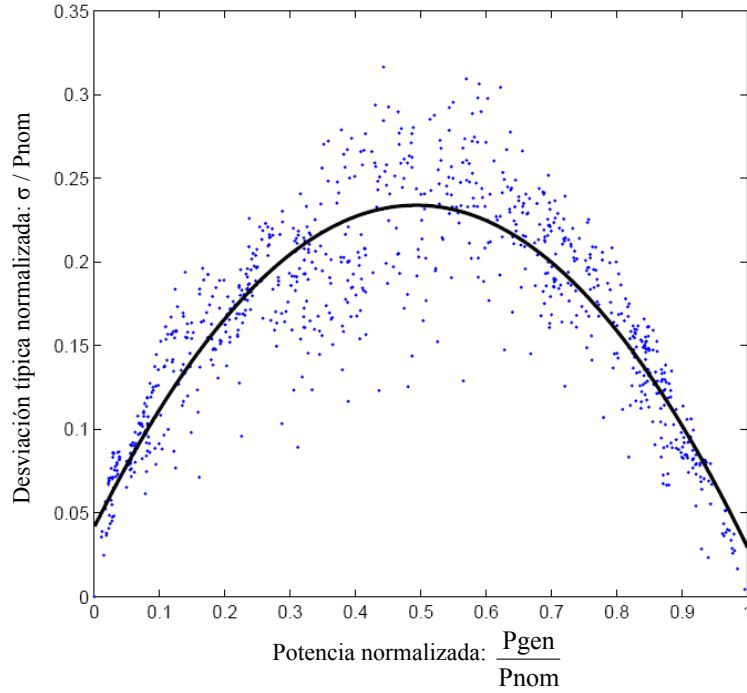


Figura 12: Relación entre la desviación típica y la potencia generada por el parque eólico.

A partir de la media, η y la varianza, σ^2 , se calculan los parámetros a y b para cada parque eólico, de la forma siguiente: [1]

$$a = \eta \left(\frac{\eta(1-\eta)}{\sigma^2} - 1 \right) \quad b = (1-\eta) \left(\frac{\eta(1-\eta)}{\sigma^2} - 1 \right)$$

Hasta ahora se ha hablado de las variables aleatorias (potencia generada por cada parque) como variables continuas e independientes. Pero la realidad del caso que nos ocupa, es que las variables aleatorias guardan una cierta relación entre sí; de manera que las potencias generadas por los parques eólicos de una misma zona geográfica seguirán un comportamiento similar, ya que estos parques se ven afectados por unas condiciones climatológicas similares y la potencia generada depende directamente de la velocidad y dirección del viento.

Por tanto a la hora de generar potencias aleatorias según la FDP beta, ha de ser tenida en cuenta esta relación entre los parques; teniendo entonces un problema de variables aleatorias continuas y dependientes.

Esta relación entre los parques de una misma zona geográfica será expresada mediante una matriz de coeficientes de correlación entre parques. Esta matriz debe ser semidefinida positiva, los elementos de la diagonal principal serán unos ya que son los coeficientes de correlación de cada parque con él mismo, el resto de elementos serán, por filas, los coeficientes de correlación entre un parque y el resto de parques de la zona. Seguidamente se muestra un ejemplo:

$$\begin{pmatrix} 1 & 0.8 & 0.5 & 0.6 & 0.6 \\ 0.8 & 1 & 0.8 & 0.6 & 0.6 \\ 0.5 & 0.8 & 1 & 0.75 & 0.75 \\ 0.6 & 0.6 & 0.75 & 1 & 0.9 \\ 0.6 & 0.6 & 0.75 & 0.9 & 1 \end{pmatrix}$$

Para generar nuevos valores de las variables aleatorias dependientes, según la FDP beta dependiente, son necesarios varios pasos previos; ya que se necesita obtener valores de probabilidad (entre 0 y 1) relacionados entre sí, según la matriz de coeficientes anterior. Con estos valores de probabilidad correlacionados y la FDP beta se generan nuevos valores de la variable aleatoria, potencia eólica.

Para ello, a partir de la matriz de correlación de los parques de una misma zona y el número de iteraciones necesario, se genera una matriz de números aleatorios según una distribución normal multivariable de media cero.

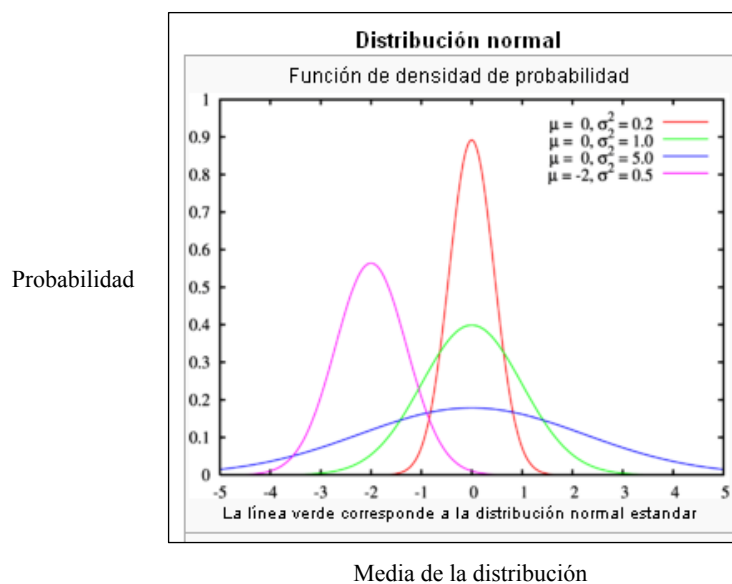
Con esta matriz de números aleatorios obtenida, se calcula mediante una FDP normal la probabilidad de ocurrencia de cada uno de estos valores y se obtiene una nueva matriz de probabilidades. [1]

Con la matriz de probabilidades y los parámetros “a” y “b” se calcula la inversa de la FDP beta, obteniéndose un valor para la variable aleatoria, potencia generada por cada parque eólico, en p.u.

De esta forma se consiguen valores aleatorios de potencia generada que siguen la relación fijada con otros parques de la misma zona, con lo que se aproxima así el comportamiento del modelo generado al real del sistema eléctrico de estudio.

5.3. Modelado del comportamiento de las cargas.

La variable, potencia consumida por una carga de un sistema eléctrico, se puede modelar como una FDP normal, figura 13, [10] cuya media será la potencia consumida por cada carga y la desviación típica será la posible variación que puede experimentar cada carga dependiendo de diferentes factores, como pueden ser: temperatura ambiental, periodo del día, día de la semana, estacionalidad, etc...



Media de la distribución
Figura 13. Distribución normal

La desviación típica de la carga podría calcularse mediante un amplio muestreo del comportamiento de determinadas cargas tipo, a lo largo del tiempo. Para realizar este muestreo deben elegirse muestras de las cargas que hayan sido afectadas de manera similar por los factores anteriormente mencionados.

Se podría proceder de la siguiente forma:

Tomando como periodo de muestreo 2 años, analizar los valores que presentó una carga determinada todos los martes, miércoles y jueves de los meses de diciembre, enero y febrero a las 9 horas de la mañana, excluyendo aquellas en las que la temperatura ambiental del día estuviese fuera del intervalo [3 °C, 10°C]. De esta forma puede calcularse la posible variabilidad de la carga.

Para este proyecto se ha tomado como desviación típica un 10% del valor medio.¹

5.4 Descripción del proceso para la resolución de un flujo de cargas probabilístico mediante el método de Monte Carlo.

Se parte de un modelo del sistema eléctrico, que en los programas de simulación de sistemas eléctricos se conoce como *caso base*. Este *caso base* contiene todos los datos de la red eléctrica (parámetros característicos de líneas, generadores, cargas, etc...).

En él están incluidos los valores de demanda y potencia eólica previstos y los valores de generación convencional programados para el escenario horario a estudiar.

Se lee el *caso base* con PSS/E y se extraen los valores de potencia eólica prevista para cada parque y la potencia nominal de cada uno de ellos. Con estos valores se calcula, para cada parque, la media, η , y la varianza, σ^2 , de una FDP beta.

$$\eta = \frac{P_{prev}}{P_{nom}}$$

$$\sigma^2 = (-0.79257 \cdot (P_{prev})^2 + 0.77991 \cdot P_{prev} + 0.042078)^2$$

Para obtener valores de probabilidad (entre 0 y 1) relacionados entre sí, según la matriz de correlación de las potencias de los parques de una zona y utilizarlos para generar valores aleatorios de potencia eólica según una FDP beta es necesario realizar los siguientes 4 pasos:

- 1º Con la media, η , la matriz de correlación de las potencias de los parques de una zona y el número de iteraciones del flujo de cargas probabilístico (dato de entrada); se genera una matriz de números aleatorios en torno a cero, la media (μ), según una distribución normal multivariable.
- 2º Mediante una distribución uniforme se calcula la probabilidad de ocurrencia de cada número aleatorio de la matriz obtenida anteriormente. Los valores de probabilidad obtenidos (entre 0 y 1) siguen la relación dada por la matriz de correlación de los parques de una zona.

¹. Dato obtenido de la página Web de Red Eléctrica de España. [8]

3° A partir de la media, η y la varianza, σ^2 , se calculan los parámetros a y b de la FDP beta, para cada parque eólico, de la forma siguiente: [1]

$$a = \eta \left(\frac{\eta(1-\eta)}{\sigma^2} - 1 \right) \quad b = (1-\eta) \left(\frac{\eta(1-\eta)}{\sigma^2} - 1 \right)$$

4° Con los parámetros a y b , y los valores de probabilidad obtenidos anteriormente se obtienen, mediante la inversa de la FDP beta dependiente; para cada parque, tantas potencias aleatorias generadas como número de iteraciones sea necesario.

Lo expuesto a continuación compone una iteración del cálculo.

Se introduce en el *caso base*, a cada parque eólico el valor de potencia obtenido para la iteración N y se reparte, en partes iguales, la variación de generación resultante de los parques eólicos, entre los generadores convencionales acoplados en el sistema que participan en la regulación secundaria y terciaria.

Para esta repartición se ha elegido un grupo de generadores térmicos convencionales que normalmente participan en la regulación secundaria y terciaria, ya que en la realidad son los generadores que participan en estos dos tipos de regulación los que asumen la variación de generación eólica.

Se podría haber elegido cualquier otra forma de repartir la variación de generación eólica resultante.

Esta asignación a la generación convencional, de la variación de potencia provocada por los parques eólicos es necesario hacerla en la simulación para que el nudo oscilante, nudo que en un flujo de cargas asume las variaciones de generación, no tenga que absorberla, ya que al ser un valor de potencia alto podría provocar que el caso no convergiera al resolver el flujo de cargas; debido a que el nudo oscilante no pudiera asumir un valor tan alto de potencia.

Seguidamente se genera un valor de potencia aleatoria consumida por cada carga, utilizando una FDP normal. La potencia a consumir prevista para cada carga será la media de la distribución; la desviación típica de la carga debe ser conocida.

Tras la generación de las potencias aleatorias consumidas por las cargas, se introducen éstas en los nudos correspondientes

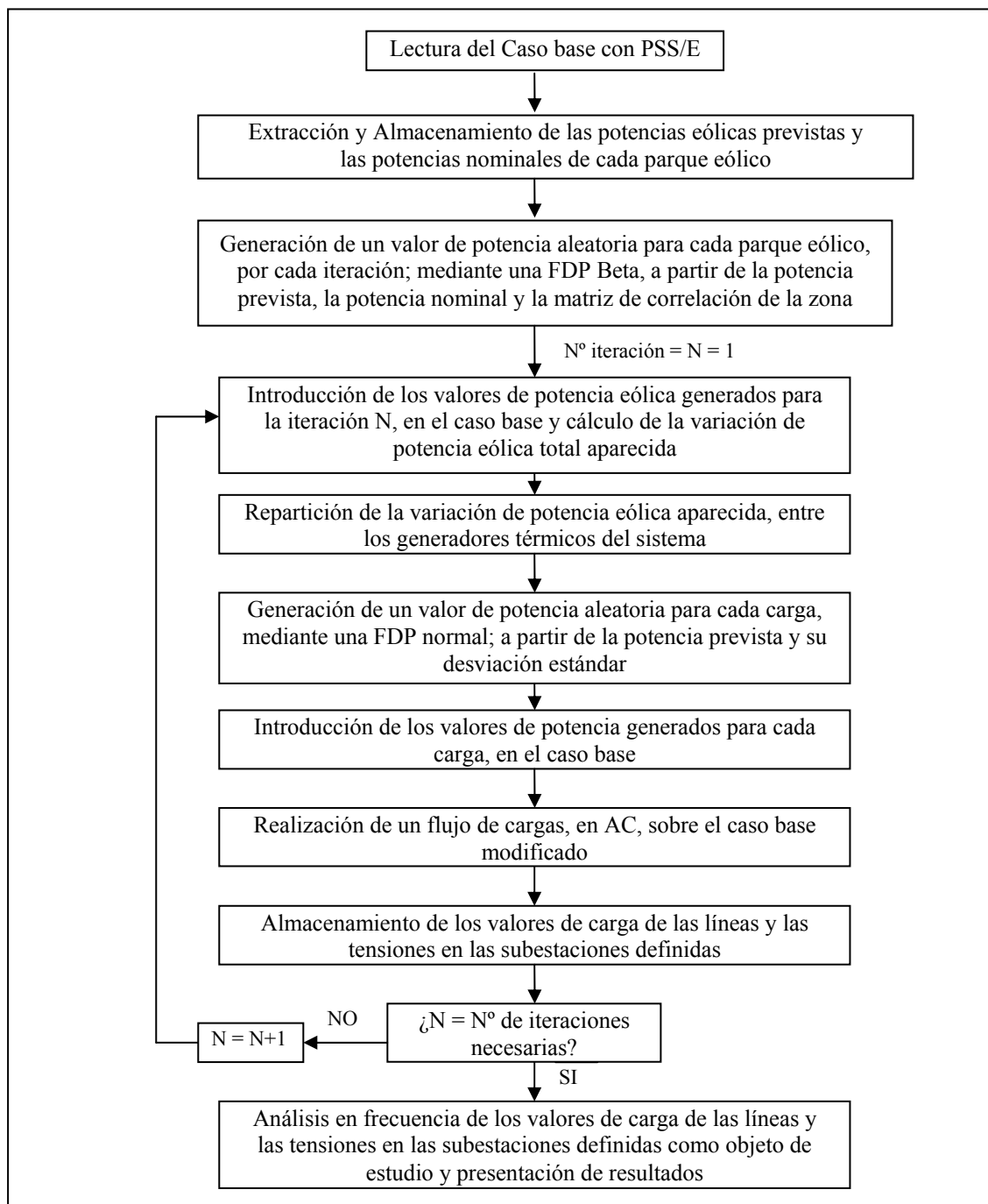
Posteriormente se resuelve un flujo de cargas en corriente alterna, ya que proporciona valores más aproximados a la realidad que un flujo de cargas en corriente continua; al considerar los valores de tensiones de los nudos, del caso y no considerar valores de tensiones nominales.

Se almacenan los valores de tensiones de las subestaciones y las cargas de las líneas, que se hayan definido como significativas por su alta probabilidad de sobrepasar sus valores nominales; ya que posteriormente serán necesarios para realizar un análisis y extraer resultados.

Todo lo expuesto hasta aquí se considera una iteración y habrá de repetirse el número de veces que se haya calculado, para que los resultados se encuentren dentro del intervalo de confianza que se desee.

Una vez realizadas las iteraciones o ejecuciones necesarias, se analizan los resultados obtenidos. Lo más habitual es realizar un análisis en frecuencia de las cargas de las líneas y las tensiones de las subestaciones. Se obtiene así, para el escenario futuro estudiado, la probabilidad de superar un cierto valor de carga en las líneas y la distribución de tensiones en las subestaciones definidas.

A continuación se muestra un flujograma con los pasos descritos.



Flujograma 1. Pasos a seguir para la resolución de un flujo de cargas probabilístico mediante el método de Monte Carlo

6. Descripción de los programas creados para la realización de un flujo de cargas probabilístico, mediante el método de Monte Carlo.

Para entender mejor el método utilizado, a continuación se describen, de manera detallada, los pasos seguidos en la elaboración de los programas que permiten realizar un flujo de cargas probabilístico.

Los programas y archivos creados son los siguientes:

1. Datos de Entrada
2. Lectura y almacenamiento de P y Pmax
3. DBD
4. Adapta datos a PSS/E
5. Cargar datos para FCP
6. FCP

En este proyecto se ha elegido trabajar con el modelo de la red de transporte eléctrico español, ya que los casos PSS/E de partida que se utilizarán son públicos y se encuentran en la página web de Red Eléctrica de España. [9]

Estos casos son los que Red Eléctrica de España utiliza para el análisis de restricciones diario que realiza el día D-1 para el día D. Genera un caso de PSS/E para cada hora del día D con los datos que a continuación se enumeran y una vez preparados los 24 casos, Red Eléctrica realiza el análisis de restricciones. A posteriori publica dichos casos en su página Web. [9]

Los casos PSS/E que se pueden encontrar en la página Web de Red Eléctrica de España [9] contienen: el modelo de la red de transporte eléctrico español (parámetros característicos de líneas, transformadores, grupos generadores, cargas, etc...), las interconexiones internacionales con Francia, Portugal y Marruecos, un modelo reducido de estos sistemas eléctricos que permite visualizar y modificar los intercambios de energía con dichos países, los datos de la demanda casada en el PBF (Programa Base Final) para el día D y la hora estudiada, la última previsión de generación solar y eólica que se tenía, el día D-1, en el momento de realizar el análisis de restricciones para el día D, los datos de generación convencional e intercambios internacionales del Programa Viable Provisional (PVP), es decir una vez realizado el análisis de restricciones.

Es importante hacer una aclaración con respecto a la versión y extensión de los casos de PSS/E que en este proyecto se ha utilizado.

La versión y extensión de los casos que se pueden encontrar en la página Web de Red Eléctrica de España [9] es la versión 29 y la extensión “.raw”.

La versión de PSS/E con la que se ha trabajado en este proyecto es la 31, ya que permite, como ya se dijo, programar en lenguaje Python [3].

No existe ningún inconveniente en leer y trabajar con casos de versiones anteriores de PSS/E en versiones posteriores de este programa. Por tanto al leer un caso de la versión 29, PSS/E lo transforma a la versión que se tenga instalada del programa, para nosotros la 31 y al guardar dicho caso, por defecto lo guarda con la versión 31, si se quiere guardar en otra versión, es necesario indicárselo explícitamente al programa. [4]

Respecto a la extensión, se han utilizado casos “.raw” y “.sav”.

- * Un caso “.raw”, está en formato de texto, ocupa mayor tamaño y PSS/E necesita unos 5 segundos para leerlo. [4]
- * Un caso “.sav”, está en formato binario, ocupa menor tamaño y PSS/E necesita sólo 1 segundo para leerlo. [4]

Esto se ha tenido en cuenta a la hora de optimizar el tiempo de cálculo del flujo de cargas probabilístico; ya que al ser necesario leer el caso base un gran número de veces (100-500), tantas como número de iteraciones sean necesarias; si se lee un caso “.raw” se incrementa el tiempo de cálculo innecesariamente.

Por esto se ha realizado la programación del proceso de manera que el caso de partida “.raw”, procedente de la página Web de Red Eléctrica de España, sólo se lea una vez, al comienzo y se guarde automáticamente como caso “.sav”. Este caso “.sav” será el que se lea en cada iteración. De esta forma el tiempo de cálculo se reduce.

Por tanto el primer paso para realizar un flujo de cargas probabilístico con los programas desarrollados en este proyecto, será almacenar en nuestro ordenador un caso sacado de la página Web de Red Eléctrica de España [9], que en adelante llamaremos “**caso base .raw**”. Puede guardarse en la dirección y con el nombre que se desee.

6.1. Descripción del programa: “**Datos de Entrada**”.

El siguiente paso ha sido crear un programa en código Python [3], para introducir en PSS/E los datos de entrada necesarios para los programas del flujo de cargas probabilístico.

El programa se ha llamado “**Datos de Entrada**” y contiene los siguientes datos:

1. La ruta y el nombre del **caso base .raw**.
2. La ruta y el nombre del **caso base .sav**.
3. La ruta y el nombre del archivo de Matlab, donde se almacenan, agrupadas por zonas geográficas, las potencias eólicas previstas para cada parque eólico de la zona y las potencias máximas de cada parque.
4. La ruta y nombres de los archivos de salida, en los que se muestran las circulaciones de las líneas y las tensiones de los nudos.
5. El número de bus y el identificador de cada generador eólico.
6. El número de bus y el identificador de cada carga.
7. Los números de bus y el identificador de los generadores térmicos entre los que se repartirá la variación de generación eólica surgida.
8. Los números de buses de las líneas en vigilancia y el identificador de cada una de ellas.
9. Los números de bus de las subestaciones en vigilancia.
10. El número de iteraciones necesarias.
11. El número de intervalos en los que se desea dividir, el rango en el cual se encuentran los valores calculados de cargas de las líneas y tensiones de las subestaciones, para realizar un análisis en frecuencia.
12. El porcentaje de carga de las líneas, del cual se quiere conocer la probabilidad de sobrepasarlo.

Este programa será el único que deberá modificar el usuario, a la hora de correr un flujo de cargas probabilístico. El formato debe respetarse, modificando sólo las rutas y nombres de los archivos y añadiendo o eliminando:

- Números de bus e identificador de parques eólicos, cargas, generación convencional.
- Número de bus e identificador de líneas y subestaciones que están en vigilancia.
- El número de iteraciones necesarias.
- El número de intervalos en los que se quiere dividir el rango de valores calculados, de cargas de las líneas y tensiones en las subestaciones.
- El valor de carga, en %, de las líneas en vigilancia, del cual se quiere conocer la probabilidad de sobrepasarlo.

En el apartado *A.1* del apéndice puede encontrarse una descripción detallada del formato del programa “Datos de Entrada”.

Una vez descrito el programa “Datos de Entrada”, pasamos a describir el siguiente programa que ha de ejecutarse en PSS/E: ***Lectura y Almacenamiento de P y Pmax*** .

6.2. Descripción del programa: “Lectura y Almacenamiento de P y Pmax” .

Este programa, escrito en lenguaje Python, se encarga de:

1. La lectura y convergencia del “caso base.raw”
2. El almacenamiento como “caso base.sav”, del “caso base.raw”, leído y convergido anteriormente.
3. La creación de 7 vectores vacíos, uno para cada zona geográfica en las que se ha dividido España; en los que se almacenarán las potencias eólicas previstas para los parques eólicos de cada zona y otros 7 vectores vacíos en los que se almacenarán las potencias máximas de cada parque.
4. La lectura y el almacenamiento, en los vectores correspondientes, de las potencias eólicas previstas para los parques eólicos de cada zona y de las potencias máximas de los mismos. Estos datos son extraídos del caso base.
5. Crear el archivo de texto “PPMAX_fecha_hora.m” y almacenar en él los valores de las potencias eólicas previstas y nominales, para que Matlab pueda leerlos y trabajar con ellos. Estos datos de potencia son los que el programa “DBD” de Matlab utilizará para generar aleatoriamente, según una FDP beta, potencias para cada parque.

En el apartado *A.2* del apéndice puede encontrarse una descripción detallada del código del programa: “Lectura y Almacenamiento de P y Pmax” .

6.3. Descripción del programa: “DBD” .

El siguiente programa que ha de utilizarse se ejecutará en MATLAB y se ha denominado “DBD”, las iniciales de Distribución Beta Dependiente.

Ha sido necesario utilizar Matlab para realizar los cálculos que seguidamente se describen ya que con lenguaje Python y en PSS/E, no es posible generar números según una FDP Beta dependiente.

Esta se refiere a la generación de potencias aleatorias para cada parque eólico pero siguiendo una correlación dada entre los parques de una misma zona geográfica, los coeficientes de correlación entre parques, se obtienen empíricamente realizando una comparación de las potencias generadas por los parques eólicos a lo largo de un periodo de tiempo determinado. La mayor influencia en los coeficientes de correlación está en la distribución de los parques eólicos, es decir su cercanía o lejanía y su emplazamiento específico. Pero también influye en cierta medida la tecnología de cada parque, no comportándose de igual manera dos parques eólicos próximos debido a las diferencias en la tecnología de sus alternadores.

De esta manera se consigue asemejar la simulación al comportamiento real de la generación eólica.

Este programa parte de la potencia prevista para cada parque, en MW y de su potencia nominal o máxima, en MW.

Calcula la potencia prevista, en p.u. (P_{prev}), la desviación típica, σ , para esa potencia prevista, según la fórmula experimental señalada anteriormente: [1]

$$\sigma = -0.79257 \cdot (P_{prev})^2 + 0.77991 \cdot P_{prev} + 0.042078$$

y su varianza, σ^2 .

Con la potencia prevista en p.u. (P_{prev}), que será la media, η , de la distribución beta y la varianza, se obtienen los valores de los parámetros a y b de una FDP beta.

Por otro lado a partir de la matriz de correlación o covarianzas de los parques de una misma zona se genera una matriz de números aleatorios en torno a cero (media de la distribución), según una distribución normal multivariable.

Después se calcula la FDP normal acumulativa de cada número aleatorio de la matriz obtenida anteriormente, obteniéndose la probabilidad de ocurrencia de cada uno de estos valores y se obtiene una nueva matriz de probabilidades.

Con la matriz de probabilidades y los parámetros “a” y “b” se calcula la inversa de la FDP beta, obteniéndose tantos valores de potencia para cada parque, en p.u., como número de iteraciones sean necesarias.

Estos valores en p.u. se transforman a potencia en MW y se almacenan para posteriormente ser introducidos en el caso base de PSS/E y realizar un flujo de cargas.

El objetivo que se pretende con este programa es: Obtener en un archivo de texto (uno para cada zona) y en forma matricial, los valores de potencia eólica, generados aleatoriamente, para los parques de cada zona.

En el apartado A.3 del apéndice se muestra detalladamente el código del programa “DBD”.

6.4. Descripción del programa “Adapta datos a PSS/E”.

Una vez que se tienen los 7 archivos de texto de las zonas de generación eólica, con las nuevas potencias generadas por cada parque, será necesario adaptar el formato de los datos para que PSS/E con lenguaje Python pueda trabajar con ellos. Para ello se ha creado el programa “Adapta datos a PSS/E” que transforma dichos archivos de texto en un archivo único con las datos de las potencias de los parques de las 7 zonas; dichas potencias en el formato matricial de Python.

El archivo creado se abrirá en PSS/E para introducir los datos al programa, con los que posteriormente se realizará el flujo de cargas probabilístico.

A este archivo de datos se le ha llamado “CARGAR_DATOS para FCP”.

En el apartado *A.4* del apéndice se describe el código del programa: “Adapta datos a PSS/E”.

6.5. Descripción del programa “FCP” , flujo de cargas probabilístico.

Este programa es el último que ha de ejecutarse en PSS/E para la realización de un flujo de cargas probabilístico.

El objetivo del programa es:

- 1.** Introducir, en el caso de partida o caso base, en cada nudo de generación eólica las potencias aleatorias calculadas anteriormente.
- 2.** Generar potencias aleatorias, mediante una FDP normal, para las cargas del sistema eléctrico e introducir en el caso de partida las potencias generadas para dichas cargas.
- 3.** Repartir las variaciones de generación eólica que surjan, entre la generación térmica acoplada en el sistema eléctrico.
- 4.** Correr un flujo de cargas una vez realizados los pasos anteriores.
- 5.** Almacenar los valores de carga de las líneas y los valores de tensión de las subestaciones, elegidas por el usuario, para su posterior análisis.
- 6.** Realizar un análisis en frecuencia de los valores de carga de las líneas para calcular la probabilidad de superar un valor de carga determinado y dividir en intervalos los valores obtenidos de tensiones en las subestaciones, para analizar que valores de tensiones se darán con mayor probabilidad.
- 7.** Mostrar al usuario, en dos archivos de texto (carga de líneas y tensiones en subestaciones), el análisis hecho con los resultados obtenidos.

En el apartado *A.5* del apéndice se describe detalladamente el código del programa “FCP” , flujo de cargas probabilístico.

7. Cálculo del número de iteraciones mínimo.

Para realizar un flujo de cargas probabilístico se ha elegido utilizar, en este proyecto, el método de Monte Carlo que realiza un muestreo aleatorio para revelar un comportamiento típico.

Para realizar este muestreo aleatorio de las cargas de las líneas y las tensiones en las subestaciones de la red de transporte eléctrico; se ha creado el programa “FCP” que modifica las potencias de los parques eólicos y las de cargas del caso de estudio, realiza un flujo de cargas del caso modificado y almacena los datos de interés. Este proceso se repite el número de veces que sea necesario para que la fiabilidad de los resultados obtenidos sea la deseada.

En este apartado se describe el proceso de cálculo de ese número de iteraciones mínimo, que debe ejecutarse el programa *FCP*, para realizar un muestreo aleatorio de las cargas de las líneas y las tensiones en las subestaciones, y que los resultados obtenidos tengan el nivel de confianza deseado.

La teoría de muestreo aleatorio se basa en las leyes fundamentales de la probabilidad. Los estadísticos han deducido la siguiente expresión que termina la probabilidad de “X” ocurrencias de un evento en “n” observaciones.

$$(p + q)^n = 1$$

p probabilidad de ocurrencia de un evento o suceso.

q (1-p) probabilidad de no ocurrencia.

n número de observaciones.

$(p + q)^n = 1$, se desarrolla por el binomio de Newton, el primer término dará la probabilidad de $X=0$, el segundo término, la de $X=1$ y así sucesivamente. La distribución de esta probabilidad se conoce como distribución binomial. La media es “np”, la varianza en “npq” y la desviación estándar es: \sqrt{npq}

A medida que “n” aumenta, la distribución binomial tiende a una distribución normal. El cálculo de la probabilidad de sobrecarga en las líneas de transporte, implica un muestreo bastante amplio, por lo que la distribución normal es una buena aproximación de la distribución binomial.

Luego es más conveniente usar:

$$\text{Media} = p \qquad \text{Desviación estándar} = \sigma_p = \sqrt{\frac{pq}{n}}$$

En un intento de estimar “p” (porcentaje real de que las cargas de las líneas eléctricas, estén por encima de un valor), se toma una muestra pequeña donde se calcula \hat{p} .

No se puede esperar que \hat{p} de cada muestra sea el verdadero valor de “p”. Pero si que la \hat{p} de una muestra se encuentre dentro del intervalo de $p \pm 2\sigma_p$ aproximadamente el 95 % de las veces, p.e.

Este intervalo $p \pm 2\sigma_p$ se conoce como intervalo de confianza y queda definido por un nivel de confianza: $100 \cdot (1 - \alpha)\%$.

Siendo α , la confiabilidad, consistencia de los resultados obtenidos.

Por tanto si Φ es la función de distribución normal estándar, el intervalo de confianza, para un nivel de confianza, $100 \cdot (1 - \alpha)\%$, será:

$$p \pm Z_{\alpha/2} \sqrt{\frac{p(1-p)}{n}}$$

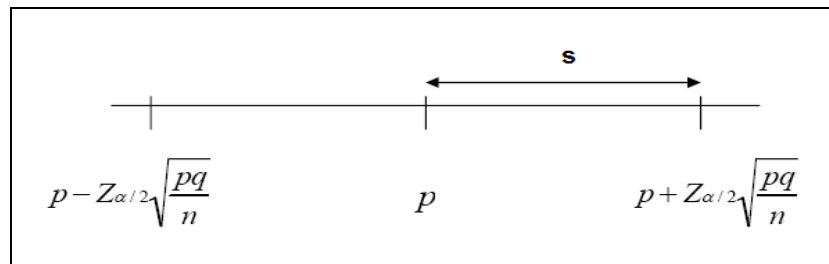


Figura 17. Representación del intervalo de confianza

s la precisión o error admisible
 $Z_{\alpha/2}$ un valor de la distribución Normal, esto es, el intervalo de confianza debe cubrir $Z_{\alpha/2}$ errores estándar, a cada lado de la media.

De modo que:

$$P(Z > Z_{\alpha/2}) = 1 - \Phi(Z_{\alpha/2}) = \alpha/2$$

$$\sigma_p = \frac{s}{Z_{\alpha/2}}$$

Por tanto para calcular el número de iteraciones necesarias se utilizará la siguiente fórmula:

$$n = \frac{Z_{\alpha/2}^2 \cdot p(1-p)}{s^2}$$

Pasos a seguir para calcular el número de iteraciones necesarias del programa FCP, para obtener los resultados con la precisión deseada.

Será necesario definir el nivel de confianza de los resultados, el error admisible de estos y un valor aproximado de probabilidad.

Se procederá de la siguiente forma:

1º Para obtener un valor aproximado de probabilidad se realizará un flujo de cargas probabilístico con el método y programas desarrollados, eligiendo por ejemplo 100 ejecuciones del programa FCP.

Se obtendrá para cada línea en vigilancia, un valor de probabilidad de superar el nivel de carga especificado, ese valor de probabilidad será el que se considere como media “p”.

2º Definir el nivel de confianza y el error admisible que se desea. Si deseamos conseguir un nivel de confianza del 95 % en que la media “p” está en un +/- 5% de la estimación (la estimación es el % de observaciones positivas), entonces se procederá del siguiente modo:

$$95\% = 100 \cdot (1 - \alpha)\%$$


$$\alpha = 0.05 \rightarrow \alpha/2 = 0.025 = 1 - \Phi(Z_{\alpha/2}) = P(Z > Z_{\alpha/2})$$

$$\Phi(Z_{\alpha/2}) = 0.975$$

Mirando en la tabla de la distribución normal estándar, figura 18. para $\Phi(Z_{\alpha/2}) = 0.975$ se tiene un valor de $Z_{\alpha/2} = 1.96$.

Esto es, el intervalo de confianza debe cubrir 1.96 errores estándar a cada lado de la media “p”.

Tabla 1. Áreas bajo la curva normal estándar. Los valores de la tabla que no se muestran en negrita representan la probabilidad de observar un valor menor o igual a z. La cifra entera y el primer decimal de z se buscan en la primera columna, y el segundo decimal en la cabecera de la tabla.

$$P(Z \leq z) = \int_{-\infty}^z f(t) dt$$


z	Segunda cifra decimal del valor de z									
	0.00	.01	.02	.03	.04	.05	.06	.07	.08	.09
0.0	.5000	.5040	.5080	.5120	.5160	.5199	.5239	.5279	.5319	.5359
0.1	.5398	.5438	.5478	.5517	.5557	.5596	.5636	.5675	.5714	.5753
0.2	.5793	.5832	.5871	.5910	.5948	.5987	.6026	.6064	.6103	.6141
0.3	.6179	.6217	.6255	.6293	.6331	.6368	.6406	.6443	.6480	.6517
0.4	.6554	.6591	.6628	.6664	.6700	.6736	.6772	.6808	.6844	.6879
0.5	.6915	.6950	.6985	.7019	.7054	.7088	.7123	.7157	.7190	.7224
0.6	.7257	.7291	.7324	.7357	.7389	.7422	.7454	.7486	.7517	.7549
0.7	.7580	.7611	.7642	.7673	.7704	.7734	.7764	.7794	.7823	.7852
0.8	.7881	.7910	.7939	.7967	.7995	.8023	.8051	.8078	.8106	.8133
0.9	.8159	.8186	.8212	.8238	.8264	.8289	.8315	.8340	.8365	.8389
1.0	.8413	.8438	.8461	.8485	.8508	.8531	.8554	.8577	.8599	.8621
1.1	.8643	.8665	.8686	.8708	.8729	.8749	.8770	.8790	.8810	.8830
1.2	.8849	.8869	.8888	.8907	.8925	.8944	.8962	.8980	.8997	.9015
1.3	.9032	.9049	.9066	.9082	.9099	.9115	.9131	.9147	.9162	.9177
1.4	.9192	.9207	.9222	.9236	.9251	.9265	.9279	.9292	.9306	.9319
1.5	.9332	.9345	.9357	.9370	.9382	.9394	.9406	.9418	.9429	.9441
1.6	.9452	.9463	.9474	.9484	.9495	.9505	.9515	.9525	.9535	.9545
1.7	.9554	.9564	.9573	.9582	.9591	.9599	.9608	.9616	.9625	.9633
1.8	.9641	.9649	.9656	.9664	.9671	.9678	.9686	.9693	.9699	.9706
1.9	.9713	.9719	.9726	.9732	.9738	.9744	.9750	.9756	.9761	.9767
2.0	.9772	.9778	.9783	.9788	.9793	.9798	.9803	.9808	.9812	.9817
2.1	.9821	.9826	.9830	.9834	.9838	.9842	.9846	.9850	.9854	.9857
2.2	.9861	.9864	.9868	.9871	.9875	.9878	.9881	.9884	.9887	.9890
2.3	.9893	.9896	.9898	.9901	.9904	.9906	.9909	.9911	.9913	.9916
2.4	.9918	.9920	.9922	.9925	.9927	.9929	.9931	.9932	.9934	.9936
2.5	.9938	.9940	.9941	.9943	.9945	.9946	.9948	.9949	.9951	.9952
2.6	.9953	.9955	.9956	.9957	.9959	.9960	.9961	.9962	.9963	.9964
2.7	.9965	.9966	.9967	.9968	.9969	.9970	.9971	.9972	.9973	.9974
2.8	.9974	.9975	.9976	.9977	.9977	.9978	.9979	.9979	.9980	.9981
2.9	.9981	.9982	.9982	.9983	.9984	.9984	.9985	.9985	.9986	.9986
3.0	.9987	.9987	.9987	.9988	.9988	.9989	.9989	.9989	.9990	.9990
3.1	.9990	.9991	.9991	.9991	.9992	.9992	.9992	.9992	.9993	.9993
3.2	.9993	.9993	.9994	.9994	.9994	.9994	.9994	.9995	.9995	.9995
3.3	.9995	.9995	.9995	.9996	.9996	.9996	.9996	.9996	.9996	.9997
3.4	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9998

Figura 18. Tabla de la distribución normal estándar

3º Una vez que se tienen todos los datos necesarios, se calcula el número de iteraciones con los valores de probabilidad de sobrecarga de cada línea obtenidos. Por lo tanto ha de calcularse tantas veces como líneas en vigilancia haya.

Los valores de probabilidad aquí mostrados, han sido elegidos al azar.

$$Z_{\alpha/2} = 1.96 \qquad n \text{ (línea 1)} = \frac{1.96^2 \cdot 0.43 \cdot 0.57}{0.05^2} = 376 \text{ iteraciones}$$

$$s = 0.05 \rightarrow 5\%$$

$$p \text{ (línea 1)} = 43 \% \qquad n \text{ (línea 2)} = \frac{1.96^2 \cdot 0.86 \cdot 0.14}{0.05^2} = 185 \text{ iteraciones}$$

$$p \text{ (línea 2)} = 86 \%$$

$$p \text{ (línea 3)} = 19 \% \qquad n \text{ (línea 3)} = \frac{1.96^2 \cdot 0.19 \cdot 0.81}{0.05^2} = 236 \text{ iteraciones}$$

4º Se elige el mayor valor obtenido como número de iteraciones mínimo para el programa *FCP*. De esta forma se asegura que los valores obtenidos de probabilidad de sobrecarga en las líneas, tendrán una fiabilidad del 95 % con un error de +/- 5 %.

8. Resultados del estudio, mediante flujo de cargas probabilístico.

En primer lugar hay que decir que para analizar correctamente la fiabilidad del flujo de cargas probabilístico mediante el método de Monte Carlo, en el estudio de las sobrecargas en las líneas de transporte eléctrico; lo ideal sería disponer, además de los casos de PSS/E con los valores de generación eólica y convencional prevista para el día D y por consiguiente los valores de carga prevista de las líneas y tensiones previstas en las subestaciones, de casos de PSS/E de tiempo real, del día D, con los valores reales de generación eólica y convencional, cargas reales de las líneas y tensiones reales en las subestaciones.

De esta manera se podrían contrastar, los resultados obtenidos con el flujo de cargas probabilístico, con los valores reales de carga de las líneas y tensiones en las subestaciones el día D y comprobar así la exactitud del método.

En este caso no es posible realizar un análisis en estas condiciones, pero si se pudiese realizar dicha comparación, la no coincidencia de los valores obtenidos con los valores reales no implicaría que el método no sea válido, ya que cualquier valores de probabilidad obtenidos es posibles, lo que reflejaría esa no coincidencia de los valores es la fiabilidad del método .

Se ha realizado una comparación de los resultados obtenidos, con datos de casos de PSS/E disponibles de horas posteriores a las del caso de estudio. Esta comparación se ha realizado a modo de curiosidad y las conclusiones obtenidas no validan o invalidan la utilización del método de flujo de cargas probabilístico para la evaluación de las sobrecargas en la red de transporte eléctrico, debidas a la generación eólica.

Otro aspecto a tener en cuenta, es que se ha comprobado que en los casos de PSS/E de la red eléctrica española que hay disponibles en la página Web de Red Eléctrica, la mayoría de los parques eólicos, aproximadamente un 80 % del total de parques instalados, están modelados como nudos de carga-generación, es decir en un mismo nudo, aparece modelado un parque eólico y una carga pero sin posibilidad de tratarlos por separado.

En momentos en los que no hay generación eólica por falta de viento, el nudo consume potencia y a medida que la generación eólica aumenta dicho nudo va dejando de consumir y pasa a generar potencia.

Estos parques modelados en nudos de carga-generación, no pueden ser utilizados en los cálculos del flujo de cargas probabilístico, ya que no cumplen la condición de que su potencia oscila entre 0 MW y su potencia nominal. Condición básica para generar potencias aleatorias mediante una FDP beta.

Este condicionante ha de ser tenido en cuenta a la hora de evaluar la validez y fiabilidad del método que aquí se presenta. Ya que la simulación que con él se realiza, del estado futuro del sistema eléctrico español, diferirá seguramente de su comportamiento real pues no se han podido tener en cuenta todos los parques eólicos instalados en el sistema eléctrico.

En determinadas zonas geográficas, en las que todos o casi todos los parques eólicos existentes estén modelados como nudos generadores independientes, los resultados obtenidos acerca de la probabilidad de sobrecarga en las líneas de la zona, si podrían ser considerados fiables y representativos del comportamiento del sistema eléctrico en esa zona y por tanto por extensión al resto del sistema, validos para evaluar la viabilidad del método propuesto.

Siguiendo este razonamiento se ha buscado en el sistema eléctrico español, la zona eléctrica y geográfica, que mayor porcentaje de parques eólicos tenga modelados como nudos generadores independientes.

Esta zona engloba los parques eólicos de: Aragón, Rioja y Navarra.

Y las líneas, que se suponen, más afectadas por la generación eólica de esta zona son líneas de 220 kV de Navarra y La Rioja:

Las líneas seleccionadas son:

L-220 kV ORCOYEN-TAFALLA
L-220 kV CORDOVILLA-SANGUESA
L-220 kV OLITE-TAFALLA
L-220 kV ORCOYEN-ICHASO, circuito1

De acuerdo con esto, se han analizado tres casos de PSS/E de la red eléctrica española, cada uno de los casos representa un escenario diferente de la red, punta de la demanda o valle, laborable o festivo.

- **Caso 1:** Corresponde a un escenario de la punta de la demanda del día 21 de abril de 2009 (martes) a las 13:36 horas.
- **Caso 2:** Corresponde a un escenario de valle de la demanda del día 22 de abril de 2009 (miércoles) a las 01:30 horas.
- **Caso 3:** Corresponde a un escenario de la punta de la demanda del día 20 de junio de 2009 (sábado) a las 10:30 horas.

Con cada uno de ellos se ha realizado un flujo de cargas probabilístico con el método propuesto y los programas creados.

A continuación se muestran, para cada caso, los resultados obtenidos de probabilidad de superar un porcentaje de carga en las líneas.

Caso 1

Corresponde a un caso de la punta de la demanda del día 21 de abril de 2009 (martes) a las 13:36 horas.

La potencia eólica real generada en el sistema eléctrico español, en ese momento, fueron 5740 MW, como puede verse en la gráfica de la figura 19, extraída de la página Web de Red Eléctrica de España. [8]

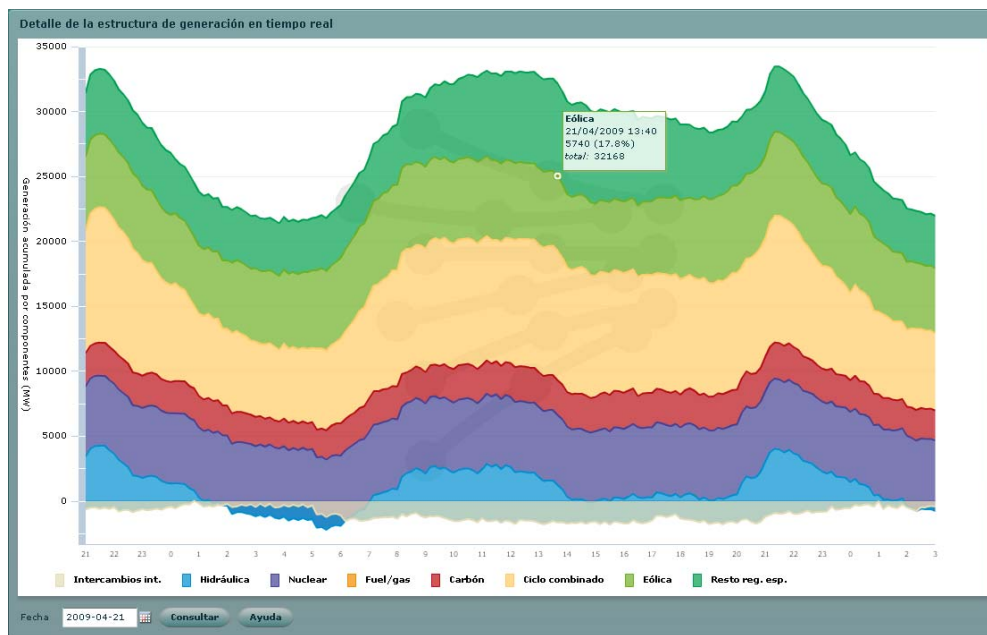


Figura 19. Distribución de la generación por tecnologías.

En el caso se ha comprobado que en la zona de: La Rioja, Navarra y Aragón, la previsión de generación eólica era elevada. Por lo tanto se han seleccionado, como se dijo antes, las líneas sobre las que se cree, más influye esta generación eólica; para analizar sus cargas y comparar los resultados obtenidos mediante un flujo de cargas probabilístico, con las cargas que previsiblemente tendrían dichas líneas unas horas más tarde.

A continuación se muestra el porcentaje de carga de cada línea en el caso de partida.

CASE F:\Proyecto USB\Prog_FCP_2009\20090421_1336.sav

ORCOYEN-TAFALLA =	92.7 %
CORDOVILLA-SANGUESA =	67.3 %
OLITE-TAFALLA =	66.6 %
ORCOYEN-ICHASO 1 =	72.9 %

A las vista de estos porcentajes parece lógico establecer como porcentaje umbral el 70 % de carga de las líneas, de manera que se calculará la probabilidad de que estas líneas superen el 70 % de su capacidad.

Para iniciar el estudio se ha elegido ejecutar el programa *FCP* 100 veces. A partir de los valores de probabilidad obtenidos se calculará el número de iteraciones mínimo, para que los resultados tengan un margen de confianza del 95 % con un error de ± 5 %, y se realizará de nuevo un flujo de cargas probabilístico ejecutando el programa *FCP* el número de veces obtenido

Una vez realizado el flujo de cargas probabilístico con 100 iteraciones se tienen los siguientes resultados:

ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\1_20090421_1336.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----

Número de ejecuciones = 100

La línea: ORCOYE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 100 %

La línea: CORDOV 1 220.00 - SANGUE 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 0 %

La línea: OLITE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 17 %

La línea: ICHASO 2 220.00 - ORCOYE 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 80 %

Con estos resultados se calcula el número de iteraciones mínimo de la siguiente forma:

$$n = \frac{Z_{\alpha/2}^2 \cdot p(1-p)}{s^2}$$

n número de observaciones.

$Z_{\alpha/2}$ un valor de la distribución Normal, esto es, el intervalo de confianza debe cubrir $Z_{\alpha/2}$ errores estándar, a cada lado de la media. En este caso 1.96

p probabilidad de ocurrencia de un evento o suceso.

$1-p$ probabilidad de no ocurrencia.

s la precisión o error admisible . En este caso un 5%

Para la línea OLITE-TAFALLA: $n = \frac{1.96^2 \cdot 0.17 \cdot 0.83}{0.05^2} = 217$ iteraciones

Para la línea ORCOYEN-ICHASO, 1: $n = \frac{1.96^2 \cdot 0.8 \cdot 0.2}{0.05^2} = 246$ iteraciones

En el caso de la línea ORCOYEN-TAFALLA no sería necesario realizar un flujo de cargas probabilístico, puesto que el porcentaje de carga en el caso de partida es muy superior al 70 % marcado, por lo tanto la posible incertidumbre en la previsión eólica no la afectaría. Ya que con un 100% de probabilidad la carga de la línea será superior al 70%.

Se elige el mayor número de iteraciones, y con 246 iteraciones, un margen de confianza del 95% y un error de +/- 5 %, se obtienen los siguientes resultados:

ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\1_20090421_1336.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----
Número de ejecuciones = 246

La línea: ORCOYE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 100 %

La línea: CORDOV 1 220.00 - SANGUE 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 0 %

La línea: OLITE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 22 %

La línea: ICHASO 2 220.00 - ORCOYE 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 84 %

A la vista de los resultados, se observa que al requerirse un número de iteraciones mínimo próximo a 100, que ha sido el valor de partida elegido para iniciar el estudio, las probabilidades con 246 iteraciones apenas varían a las calculadas inicialmente.

A modo de curiosidad se muestra el porcentaje de carga que previsiblemente tendrían las líneas horas después.

- 2 horas más tarde, en el caso de PSS/E de las 15:36 h:

ORCOYEN-TAFALLA =	84.2 %
CORDOVILLA-SANGUESA =	67.4 %
OLITE-TAFALLA =	58.5 %
ORCOYEN-ICHASO 1=	69.4 %

- 6 horas más tarde, en el caso de PSS/E de las 19:36 h:

ORCOYEN-TAFALLA =	74.2 %
CORDOVILLA-SANGUESA =	66.7 %
OLITE-TAFALLA =	50.6 %
ORCOYEN-ICHASO 1=	64.1 %

Caso 2

Corresponde a un caso del valle de la demanda del día 22 de abril de 2009 (miércoles) a las 01:30 horas.

La potencia eólica real generada en el sistema eléctrico español, en ese momento, fueron 5344 MW, como puede verse en la siguiente gráfica de la figura 20. [8]

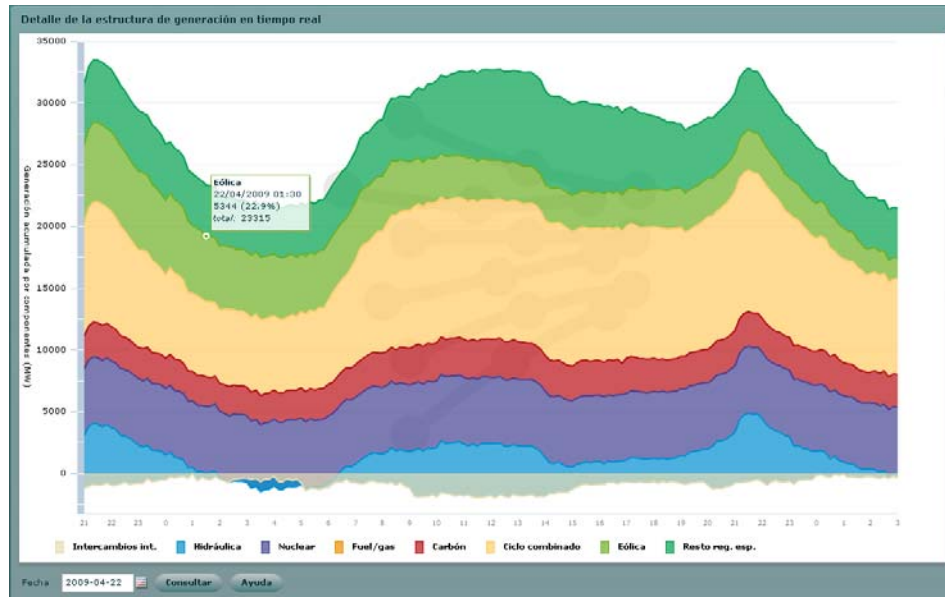


Figura 20. Distribución de la generación por tecnologías.

En el caso se ha comprobado que en la zona de: La Rioja, Navarra y Aragón, la previsión de generación eólica era elevada.

A continuación se muestra el porcentaje de carga de cada línea en el caso de partida.

CASE F:\Proyecto USB\Prog_FCP_2009\20090422_0130.sav

ORCOYEN-TAFALLA =	65.8 %
CORDOVILLA-SANGUESA =	67.5 %
OLITE-TAFALLA =	44.6 %
ORCOYEN-ICHASO 1=	69.6 %

A la vista de estos porcentajes parece lógico establecer como porcentaje umbral el 70 % de carga de las líneas, de manera que se calculará la probabilidad de que estas líneas superen el 70 % de su capacidad.

Para iniciar el estudio se elige, de nuevo, ejecutar el programa *FCP* 100 veces. A partir de los valores de probabilidad obtenidos se calculará el número de iteraciones mínimo, para que los resultados tengan un margen de confianza del 95 % con un error de +/- 5 %, y se realizará de nuevo un flujo de cargas probabilístico ejecutando el programa *FCP* el número de veces obtenido.

Una vez realizado el flujo de cargas probabilístico con 100 iteraciones se tienen los siguientes resultados:

ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\20090422_0130.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----
Número de ejecuciones = 100

La línea: ORCOYE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 13 %

La línea: CORDOV 1 220.00 - SANGUE 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 0 %

La línea: OLITE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 0 %

La línea: ICHASO 2 220.00 - ORCOYE 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 49 %

Con estos resultados se calcula el número de iteraciones mínimo de la siguiente forma:

$$n = \frac{Z_{\alpha/2}^2 \cdot p(1-p)}{s^2}$$

n número de observaciones.

$Z_{\alpha/2}$ un valor de la distribución Normal, esto es, el intervalo de confianza debe cubrir $Z_{\alpha/2}$ errores estándar, a cada lado de la media. En este caso 1.96

p probabilidad de ocurrencia de un evento o suceso.

$1-p$ probabilidad de no ocurrencia.

s la precisión o error admisible . En este caso un 5%

Para la línea ORCOYEN-TAFALLA: $n = \frac{1.96^2 \cdot 0.13 \cdot 0.87}{0.05^2} = 174$ iteraciones

Para la línea ORCOYEN-ICHASO, 1: $n = \frac{1.96^2 \cdot 0.49 \cdot 0.51}{0.05^2} = 385$ iteraciones

Se elige el mayor número de iteraciones, y con 385 iteraciones, un margen de confianza del 95% y un error de +/- 5 %, se obtienen los siguientes resultados:

ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\20090422_0130.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----
Número de ejecuciones = 385

La línea: ORCOYE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 22 %

La línea: CORDOV 1 220.00 - SANGUE 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 0 %

La línea: OLITE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 0 %

La línea: ICHASO 2 220.00 - ORCOYE 1 220.00, ckt: 1, cargará por encima del: 70 %, con una probabilidad del: 56 %

A modo de curiosidad se muestra el porcentaje de carga que previsiblemente tendrían las líneas horas después.

- 2 horas más tarde, en el caso de PSS/E de las 03:30 h:

ORCOYEN-TAFALLA =	67.7 %
CORDOVILLA-SANGUESA =	65.6 %
OLITE-TAFALLA =	43.3 %
ORCOYEN-ICHASO 1=	74.5 %

- 5 horas más tarde, en el caso de PSS/E de las 06:30 h:

ORCOYEN-TAFALLA =	73.5 %
CORDOVILLA-SANGUESA =	65.5 %
OLITE-TAFALLA =	59.3 %
ORCOYEN-ICHASO 1=	68.9 %

Caso 3

Corresponde a un caso de la punta de la demanda del día 20 de Junio de 2009 (sábado) a las 10:30 horas.

La potencia eólica real generada en el sistema eléctrico español, en ese momento, fueron 5901 MW, como puede verse en la siguiente gráfica de la figura 21. [8]

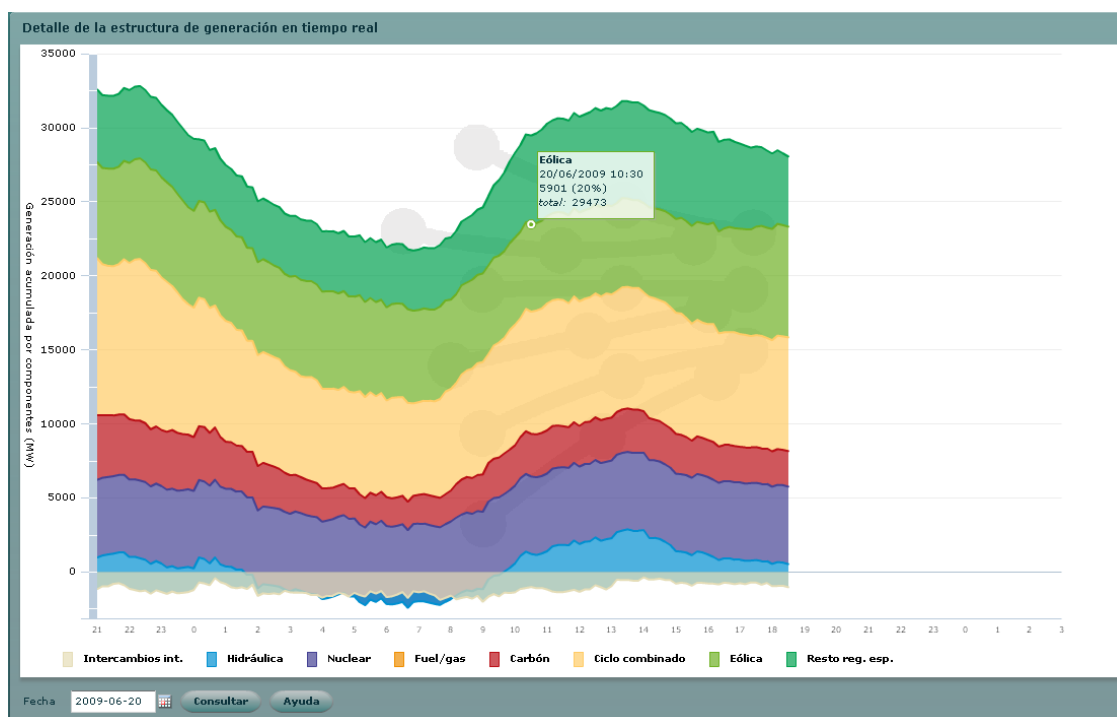


Figura 21. Distribución de la generación por tecnologías.

En el caso se ha comprobado que en la zona de: La Rioja, Navarra y Aragón, la previsión de generación eólica era elevada.

Se han modificado las líneas a vigilar ya que la distribución de la generación eólica en este caso, afectaba a otras líneas diferentes a las anteriores.

Se han añadido las líneas: L-220 kV Puentelearrá-Miranda y L-220 kV Miranda- Laguardia.

Y se ha eliminado la línea de 220 kV Olite-Tafalla.

A continuación se muestra el porcentaje de carga de cada línea en el caso de partida.

ORCOYEN-TAFALLA =	62.2 %
CORDOVILLA-SANGUESA =	82.6 %
ORCOYEN-ICHASO 1=	77.9 %
PUENTELEARRA-MIRANDA =	70.7 %
MIRANDA-LAGUARDIA =	70.9 %

A la vista de estos porcentajes de carga se han realizado dos estudios.

Uno calculando la probabilidad de que las líneas superen el 70 % de su capacidad y otro calculando la probabilidad de que las líneas superen el 80 % de su capacidad.

Para iniciar ambos estudios se elige, de nuevo, ejecutar el programa *FCP* 100 veces. A partir de los valores de probabilidad obtenidos se calculará el número de iteraciones mínimo, para que los resultados tengan un margen de confianza del 95 % con un error de +/- 5 %, y se realizará de nuevo un flujo de cargas probabilístico ejecutando el programa *FCP* el número de veces obtenido.

Una vez realizado el flujo de cargas probabilístico con 100 iteraciones y con un porcentaje umbral del **70 %** de la capacidad de las líneas, se tienen los siguientes resultados:

ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\20090620_1030.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----
Número de ejecuciones = 100

La línea: ORCOYE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 1 %
La línea: CORDOV 1 220.00 - SANGUE 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 100 %
La línea: ICHASO 2 220.00 - ORCOYE 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 95 %
La línea: PUENTE 1 220.00 - MIRAND 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 58 %
La línea: MIRAND 1 220.00 - LAGUAR 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 58 %

Con estos resultados se calcula el número de iteraciones mínimo de la siguiente forma:

$$n = \frac{Z_{\alpha/2}^2 \cdot p(1-p)}{s^2}$$

n número de observaciones.

$Z_{\alpha/2}$ un valor de la distribución Normal, esto es, el intervalo de confianza debe cubrir $Z_{\alpha/2}$ errores estándar, a cada lado de la media. En este caso es 1.96

p probabilidad de ocurrencia de un evento o suceso.

$1-p$ probabilidad de no ocurrencia.

s la precisión o error admisible . En este caso un 5%

Para la línea ICHASO-ORCOYEN:
$$n = \frac{1.96^2 \cdot 0.95 \cdot 0.05}{0.05^2} = 73 \text{ iteraciones}$$

Para la línea PUENTE LARRÁ-MIRANDA Y MIRANDA-LAGUARDIA:

$$n = \frac{1.96^2 \cdot 0.58 \cdot 0.42}{0.05^2} = 375 \text{ iteraciones}$$

Se elige el mayor número de iteraciones, y con 375 iteraciones, un margen de confianza del 95% y un error de +/- 5 %, se obtienen los siguientes resultados:

ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\20090620_1030.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----
Numero de ejecuciones = 375

La línea: ORCOYE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 0 %
La línea: CORDOV 1 220.00 - SANGUE 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 99 %
La línea: ICHASO 2 220.00 - ORCOYE 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 96 %
La línea: PUENTE 1 220.00 - MIRAND 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 57 %
La línea: MIRAND 1 220.00 - LAGUAR 1 220.00, ckt: 1, cargara por encima del: 70 %, con una probabilidad del: 59 %

Para calcular la probabilidad de que las líneas superen el **80 %** de su capacidad, se comienza de nuevo ejecutando el programa *FCP* 100 veces y se tienen los siguientes resultados:

ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\20090620_1030.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----
Numero de ejecuciones = 100

La línea: ORCOYE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 0 %
La línea: CORDOV 1 220.00 - SANGUE 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 82 %
La línea: ICHASO 2 220.00 - ORCOYE 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 34 %
La línea: PUENTE 1 220.00 - MIRAND 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 1 %
La línea: MIRAND 1 220.00 - LAGUAR 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 1 %

Se calcula el número de iteraciones mínimo con estos resultados:

Para la línea CORDOVILLA-SANGÜESA:
$$n = \frac{1.96^2 \cdot 0.82 \cdot 0.18}{0.05^2} = 227 \text{ iteraciones}$$

Para la línea ICHASO-ORCOYEN, 1:
$$n = \frac{1.96^2 \cdot 0.34 \cdot 0.66}{0.05^2} = 345 \text{ iteraciones}$$

Para la línea PUENTE LARRÁ-MIRANDA Y MIRANDA-LAGUARDIA:

$$n = \frac{1.96^2 \cdot 0.01 \cdot 0.01}{0.05^2} = 16 \text{ iteraciones}$$

Se elige el mayor número de iteraciones, y con 345 iteraciones, un margen de confianza del 95% y un error de +/- 5 %, se obtienen los siguientes resultados:

ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\20090620_1030.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----
Numero de ejecuciones = 345

La línea: ORCOYE 1 220.00 - TAFALL 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 0 %
La línea: CORDOV 1 220.00 - SANGUE 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 83 %
La línea: ICHASO 2 220.00 - ORCOYE 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 31 %
La línea: PUENTE 1 220.00 - MIRAND 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 0 %
La línea: MIRAND 1 220.00 - LAGUAR 1 220.00, ckt: 1, cargara por encima del: 80 %, con una probabilidad del: 1 %

A modo de curiosidad se muestra el porcentaje de carga que previsiblemente tendrían las líneas horas después.

- 2 horas más tarde, en el caso de PSS/E de las 12:36 h:

ORCOYEN-TAFALLA =	63.5 %
CORDOVILLA-SANGUESA =	85.1 %
ORCOYEN-ICHASO 1=	80.7 %
PUENTELARRA-MIRANDA =	78.9 %
MIRANDA-LAGUARDIA =	81.8 %

- 5 horas más tarde, en el caso de PSS/E de las 15:26 h:

ORCOYEN-TAFALLA =	58.7 %
CORDOVILLA-SANGUESA =	94.6 %
ORCOYEN-ICHASO 1=	87.9 %
PUENTELARRA-MIRANDA =	85.2 %
MIRANDA-LAGUARDIA =	88.8 %

Resultados de la distribución de la tensión en las subestaciones.

A continuación se muestran los resultados obtenidos, con el caso 3, sobre los valores más probables de tensiones, que se darían en las subestaciones.

A la vista de los resultados se observan variaciones de tensión muy pequeñas en torno a 1 ó 2 kV.

ARCHIVO DE SALIDA PARA LAS TENSIONES DE LAS SUBESTACIONES

Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\20090620_1030.sav

----- ANALISIS EN FRECUENCIA DE LAS TENSIONES DE LAS SUBESTACIONES -----
Numero de ejecuciones = 345

ANALISIS EN FRECUENCIA DE LA TENSION DE LA SUBESTACION: P.G.RO 1 400.00

Numero de veces que la tensión de la subestación estará en el intervalo:

410.132 kV - 410.295 kV: 13
410.295 kV - 410.458 kV: 29
410.458 kV - 410.621 kV: 66
410.621 kV - 410.784 kV: 94
410.784 kV - 410.947 kV: 109
410.947 kV - 411.110 kV: 30
411.110 kV - 411.273 kV: 1
411.273 kV - 411.436 kV: 0
411.436 kV - 411.599 kV: 1
411.599 kV - 411.762 kV: 2

ANALISIS EN FRECUENCIA DE LA TENSION DE LA SUBESTACION: MESON 2 220.00

Numero de veces que la tensión de la subestación estará en el intervalo:

223.188 kV - 223.311 kV: 41
223.311 kV - 223.434 kV: 170
223.434 kV - 223.558 kV: 125
223.558 kV - 223.681 kV: 6
223.681 kV - 223.804 kV: 0
223.804 kV - 223.927 kV: 0
223.927 kV - 224.050 kV: 0
224.050 kV - 224.173 kV: 0
224.173 kV - 224.296 kV: 1
224.296 kV - 224.419 kV: 2

ANALISIS EN FRECUENCIA DE LA TENSION DE LA SUBESTACION: TAFALL 1 220.00

Numero de veces que la tensión de la subestación estará en el intervalo:

221.063 kV - 221.084 kV: 5
221.084 kV - 221.105 kV: 12
221.105 kV - 221.126 kV: 26
221.126 kV - 221.146 kV: 55
221.146 kV - 221.167 kV: 71
221.167 kV - 221.188 kV: 57
221.188 kV - 221.209 kV: 46
221.209 kV - 221.229 kV: 38
221.229 kV - 221.250 kV: 28
221.250 kV - 221.271 kV: 7

ANALISIS EN FRECUENCIA DE LA TENSION DE LA SUBESTACION: MONTET 1 220.00

Numero de veces que la tensión de la subestación estará en el intervalo:

227.094 kV - 227.124 kV: 1
227.124 kV - 227.153 kV: 8
227.153 kV - 227.183 kV: 15
227.183 kV - 227.212 kV: 27
227.212 kV - 227.242 kV: 61
227.242 kV - 227.271 kV: 75
227.271 kV - 227.301 kV: 82
227.301 kV - 227.330 kV: 53
227.330 kV - 227.360 kV: 19
227.360 kV - 227.389 kV: 4

Resultado del tiempo necesario para realizar el flujo de cargas probabilístico.

Tiempo empleado en realizar el análisis completo de cada caso, para calcular la probabilidad de superar un porcentaje de carga en determinadas líneas

- En el análisis completo del caso 1 se tardó unos 13 minutos, la mayor parte, 9 minutos, los consume el programa *FCP*, ya que debe ejecutarse 246 veces; el resto del tiempo lo consume la ejecución del resto de programas.
- En el análisis completo del caso 2 se tardó unos 15 minutos, la mayor parte, 11 minutos, los consume el programa *FCP* al ejecutarse 385 veces; el resto del tiempo lo consume la ejecución del resto de programas.
- En el análisis completo del caso 3 se tardó unos 14 minutos, la mayor parte, 10 minutos, los consume el programa *FCP* al ejecutarse 375 veces; el resto del tiempo lo consume la ejecución del resto de programas.

Estos valores se muestran en la tabla 1:

<i>Casos</i>	<i>Nº de iteraciones necesarias</i>	<i>Tiempo total necesario para su análisis (minutos)</i>	<i>Tiempo necesitado por el programa FCP</i>	<i>Tiempo necesitado por el resto de programas</i>
Caso 1	246	13	9	4
Caso 2	385	15	11	4
Caso 3	375	14	10	4

Tabla 1. Numero de iteraciones y tiempo de cálculo necesarios.

9. Conclusiones.

Como se dijo en el apartado anterior, para comprobar la fiabilidad del método propuesto, los resultados obtenidos deberían ser comparados con los valores de carga reales que tuvieron las líneas, el día del caso estudiado y en la hora analizada. Como no se dispone de estos datos reales se ha realizado una comparación, a modo de curiosidad, con los valores de carga de las líneas que aparecen en los casos de PSS/E de horas posteriores a la analizada, disponibles en la Web de Red Eléctrica de España; estos valores de carga de las líneas son los que previsiblemente tendrían las líneas el día estudiado y en la hora analizada, al ser calculados a partir de las previsiones de generación eólica y convencional que se tenían el día anterior al del caso.

También ha de ser tenido en cuenta que en los casos de PSS/E disponibles del sistema eléctrico español, la mayoría de los parques eólicos, aproximadamente un 80 % del total de parques instalados, están modelados como nudos de carga-generación y no pueden ser utilizados en los cálculos del flujo de cargas probabilístico, ya que no cumplen la condición de que su potencia oscila entre 0 MW y su potencia nominal. Condición básica para generar potencias aleatorias mediante una FDP beta. Por tanto solo se están considerando el 20 % de los parques eólicos instalados.

Estas dos particularidades han de ser tenidas en cuenta a la hora de evaluar la fiabilidad del método que aquí se presenta. Ya que la simulación que con él se realiza, del estado futuro del sistema eléctrico español, diferirá seguramente de su comportamiento real; pues no se han podido tener en cuenta todos los factores que influyen en el comportamiento del sistema eléctrico y los datos con los que se comparan los resultados no reflejan el comportamiento real del sistema eléctrico.

A continuación se muestra, a modo de curiosidad, un análisis de los resultados obtenidos:

A la vista de los resultados al analizar el caso 1, se puede decir que para realizar una previsión a corto plazo este método resulta útil, puesto que la línea Olite-Tafalla, para la que se ha obtenido un 22% de probabilidad de que superase el 70% de su capacidad, no lo superó 2 horas después a la del caso estudiado y la línea Orcoyen-Ichaso para la que se ha obtenido un 84% de probabilidad de que superase el 70% de su capacidad, 2 horas después estaba al 70% de su carga.

Si la previsión se realiza para un horizonte temporal más amplio, los resultados reflejan que la fiabilidad del método disminuye, puesto que la línea Orcoyen-Ichaso que tenía una alta probabilidad de superar el 70% de su carga, no lo superó.

Los resultados obtenidos al analizar el caso 2, muestran que para realizar una previsión a corto plazo el método es válido, ya que la evolución de las cargas de las líneas Orcoyen-Tafalla y Orcoyen-Ichaso validan los valores de probabilidad obtenidos, sin embargo en este caso se observa que si nos guiásemos por los resultados de probabilidad obtenidos para realizar una previsión sobre un horizonte temporal mayor, 5 horas; nos estaríamos equivocando a la hora de decir, que es poco probable que la línea Orcoyen-Tafalla supere el 70% de su carga.

Con respecto a los resultados obtenidos al analizar el caso 3, se puede decir que los valores de probabilidad calculados se confirman con las cargas de las líneas 2 horas más tarde. Por tanto en este caso también se considera válido el método utilizado para realizar previsiones a corto plazo.

Para previsiones a largo plazo el método no es tan fiable, como se comprueba con la línea L-220 kV Orcoyen-Ichaso 1, ya que la probabilidad de superar el 80% de su capacidad es del 31% y 5 horas después del caso de estudio, la línea estaba al 88% de su capacidad.

Como conclusión final se puede decir que para realizar previsiones a corto plazo (2, 3 horas) los resultados que aporta el método son fiables, sin embargo a medida que el horizonte temporal de las previsiones aumenta, la fiabilidad de los resultados disminuye, como es lógico por otra parte.

Los valores obtenidos de tensiones en las subestaciones no pueden ser valorados de forma adecuada con la información de que se dispone. Ya que sólo Red Eléctrica de España, como operador del sistema, conoce los valores de tensiones que se consideran normales en cada subestación o zona eléctrica.

Estos resultados servirían para hacerse una idea de cómo evolucionarían las tensiones en las distintas subestaciones y conociendo los márgenes de tensión en los que cada subestación debe estar, identificar posibles problemas de sobretensiones o subtensiones.

Se han observado variaciones de tensión muy pequeñas en torno a 1 ó 2 kV, esto se debe a que en el flujo de cargas probabilístico no se modifican las potencias reactivas de los generadores eólicos, y por tanto las variaciones de tensión en las subestaciones se deben solamente a las variaciones de carga por las líneas; debidas éstas a la variación de la potencia demandada y de la generación eólica.

La realidad en este tema es distinta, ya que los generadores eólicos por norma general al ser máquinas de inducción, consumen potencia reactiva al aumentar la potencia activa generada y el consumo de potencia reactiva baja las tensiones. Modelar este comportamiento de los aerogeneradores no resulta sencillo, ya que no se dispone de las curvas que representan la potencia reactiva en función de la activa.

Por tanto los resultados obtenidos con respecto a las tensiones, no pueden ser considerados válidos para predecir el comportamiento de éstas en un escenario futuro.

Respecto al tiempo empleado en realizar un estudio o análisis completo de un caso, puede verse en la tabla 1 como el tiempo aumenta, al aumentar el número de iteraciones necesarias del programa FCP.

<i>Casos</i>	<i>Nº de iteraciones necesarias</i>	<i>Tiempo total necesario para su análisis (minutos)</i>	<i>Tiempo necesitado por el programa FCP</i>	<i>Tiempo necesitado por el resto de programas</i>
Caso 1	246	13	9	4
Caso 2	385	15	11	4
Caso 3	375	14	10	4

Tabla 1. Numero de iteraciones y tiempo de cálculo necesarios.

El número de iteraciones puede disminuirse, y por tanto el tiempo de cálculo, a costa de disminuir la fiabilidad de los resultados y aumentar el error de éstos.

El tiempo necesario para el análisis no evoluciona linealmente, se comprueba que para 100 ejecuciones del programa *FCP*, se necesitan unos 2 minutos y medio y para algo más del doble de ejecuciones son necesarios 9 minutos. El tiempo necesario para realizar las últimas 10 iteraciones en el caso 2, ha sido de 1 minuto; comparándolo con el tiempo necesario para el caso 3.

Con el caso 1 se han realizado pruebas con 500 y 1000 iteraciones y el tiempo necesario ha sido notablemente alto; con 500 iteraciones 19 minutos y con 1000 iteraciones 44 minutos.

Por lo que se deduce que la evolución del tiempo necesario del programa *FCP* no aumenta de forma proporcional al aumentar su número de ejecuciones.

En la tabla 2 se muestran los valores anteriores:

Nº de ejecuciones del programa <i>FCP</i>	Tiempo necesario por el programa <i>FCP</i> (minutos)
100	2,5
246	9
500	19
1000	44

Tabla 2. Número de ejecuciones del programa *FCP* y tiempo necesario para el caso 1.

En el gráfico 1 se han representado los datos de la tabla 2, puede verse como el tiempo necesario no aumenta de forma proporcional al número de ejecuciones necesarias.

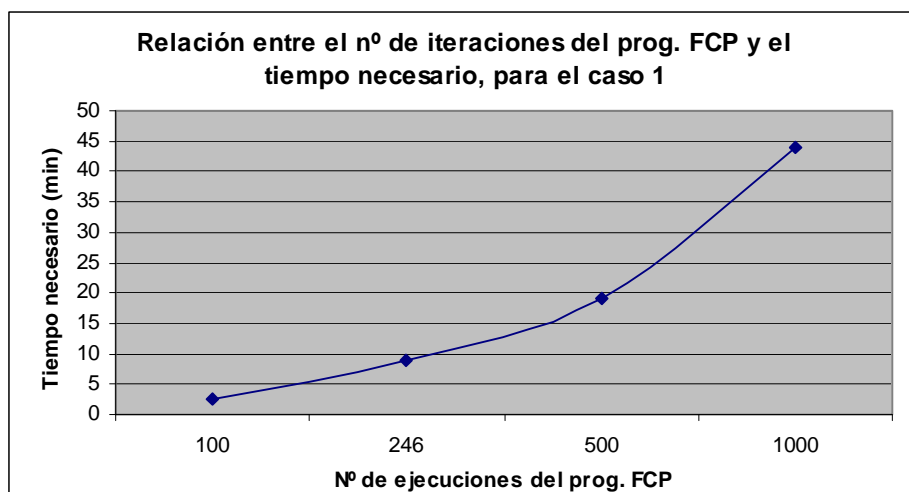


Gráfico 1. Relación entre el nº de ejecuciones del programa FCP para el caso 1 y el tiempo necesario

Es muy recomendable cada vez que se quiera analizar un nuevo caso, si se acaba de analizar otro anteriormente, cerrar y volver a abrir el programa PSS/E; ya que este se vuelve más lento a medida que almacena datos en su memoria interna y el tiempo de cálculo aumenta considerablemente. De esta manera al volver a iniciarlo se vacía su memoria interna y el tiempo de cálculo no aumenta innecesariamente.

Con todo lo anterior puede decirse que este método puede ser útil al operador del sistema eléctrico español, Red Eléctrica de España, a la hora de realizar los distintos análisis de restricciones el día D para el día D+1 o el mismo día D+1.

A medida que nos acercamos en el tiempo al escenario estudiado, las previsiones de generación eólica son más precisas, por lo que los resultados obtenidos con este método deberían ser más fiables.

Anexo.

A.1. Pasos a seguir por el usuario para ejecutar un flujo de cargas probabilístico.

Para poder ejecutar un flujo de cargas probabilístico es necesario tener instalado en el ordenador que se vaya a ejecutar: PSS/E, versión 31 y Matlab, la versión es indiferente.

Lo más cómodo para el usuario es copiar todos los programas y archivos necesarios en una misma carpeta; esta carpeta podrá ubicarse en cualquier directorio del ordenador. Si los programas se encuentran en un dispositivo de almacenamiento portátil no será necesario copiarlos a la memoria del ordenador, pueden ejecutarse desde dicho dispositivo.

Los programas y archivos necesarios son:

- Datos de Entrada
- Lectura y almacenamiento de P y Pmax
- DBD
- Adapta datos a PSS/E
- CARGAR_DATOS para FCP
- FCP

Por comodidad, es conveniente guardar en la misma carpeta en la que se encuentren los programas, los casos de PSS/E con los que se va a trabajar.

Los pasos a seguir para la ejecución de un flujo de cargas probabilístico son los siguientes:

1º Introducir los datos de entrada en el programa: “Datos de Entrada”. Para ello el usuario introducirá la ruta y nombre del caso PSS/E con el que quiere trabajar, tanto del caso “.raw” como del “.sav”. Debe hacer lo mismo con el archivo de almacenamiento de potencias de los parques eólicos, “PPMAX_fecha_hora.m” y los archivos de salida, “SALIDALIN_fecha_hora.txt” y “SALIDATEN_fecha_hora.txt”.

Es importante destacar que los nombres de estos archivos deberán escribirse con su correspondiente extensión; en los dos primeros debe dejarse un espacio en blanco antes y después de las triples comillas pero en los tres últimos no debe dejarse ningún espacio en blanco antes y después de las comillas y debe respetarse siempre el formato con que se encuentran escritos en el archivo, de lo contrario los programas no se podrán ejecutar. Se muestra como deben escribirse.

```
CASOraw= r" F:\Proyecto USB\Prog_FCP_2009\20090421_1336.raw ""
CASOsav= r" F:\Proyecto USB\Prog_FCP_2009\20090421_1336.sav ""
PzonasPPMAXzonas= "F:\Proyecto USB\Prog_FCP_2009\PPMAX_20090421_1336.m"
SALIDALINEA= "F:\Proyecto USB\Prog_FCP_2009\SALIDALIN_20090421_1336.txt"
SALIDATENSION= "F:\Proyecto USB\Prog_FCP_2009\SALIDATEN_20090421_1336.txt"
```

También modificará a su elección los números de bus y los identificadores de:

- Los parques eólicos, respetando la clasificación hecha por zonas geográficas. De manera que si se desea añadir un parque a los existentes se introducirá en el vector de parques de la zona a la que corresponda. Es importante recordar que el identificador de los elementos ha de ir entre comillas simples.

- Cargas
- Generadores térmicos
- Líneas a vigilar
- Subestaciones de las que se quiere analizar la tensión

Otros datos a modificar son:

- El número de iteraciones necesarias para la realización del flujo de cargas probabilístico.
- El número de intervalos en los que se desea dividir el rango de los valores obtenidos de carga en las líneas y de tensiones en las subestaciones.
- El porcentaje de capacidad de las líneas, del que se quiere conocer la probabilidad de superarlo.

Después de realizar las modificaciones necesarias en este programa, se guardan los cambios y se cierra.

2º Seguidamente debe abrirse PSS/E y pinchar en el botón “CLI”, que se encuentra en las barras de herramientas superiores. Si no se encuentra en dichas barras de herramientas, se buscará en el menú “I/O Control” de la barra de comandos superior, según se muestra en la figura 14.

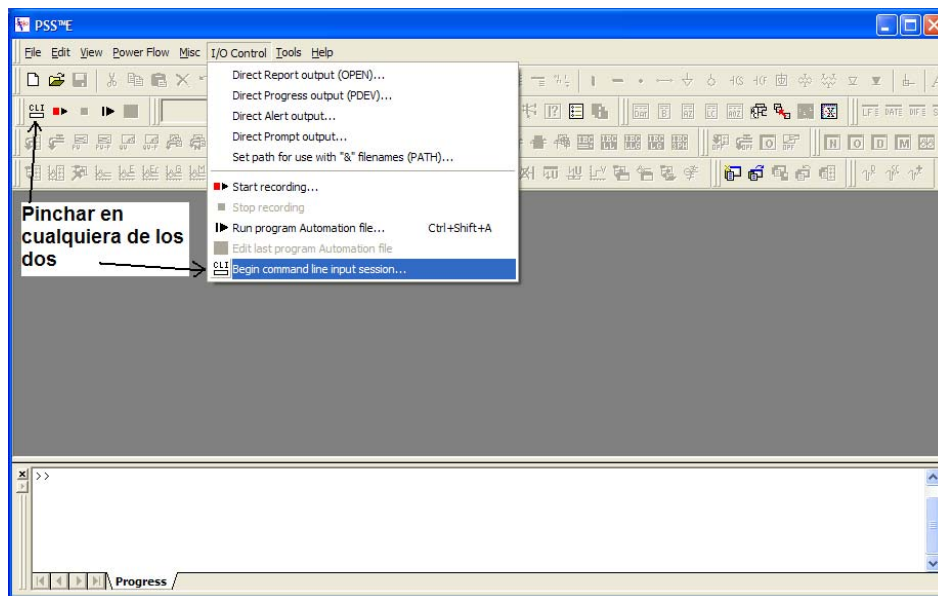


Figura 14. Pantalla de inicio de PSS/E

Se abrirá una nueva ventana “Command Line Input” donde se pueden introducir instrucciones o comandos para ser ejecutados por PSS/E.

En este caso debe elegirse que el lenguaje de los comandos que se van ejecutar sea Python. Para ello en la parte inferior derecha de esta ventana aparece un recuadro desplegable “Command language” en el que se elegirá “Python”.

Hecho esto ya se estará en condiciones de ejecutar los programas del flujo de cargas probabilístico. Pinchando en el botón “Run Auto” de esta ventana aparece una nueva ventana, figura 15, en la que se buscará el programa “Datos de Entrada. py”, ha de seleccionarse en tipo de archivo: “Python file” y se abrirá.

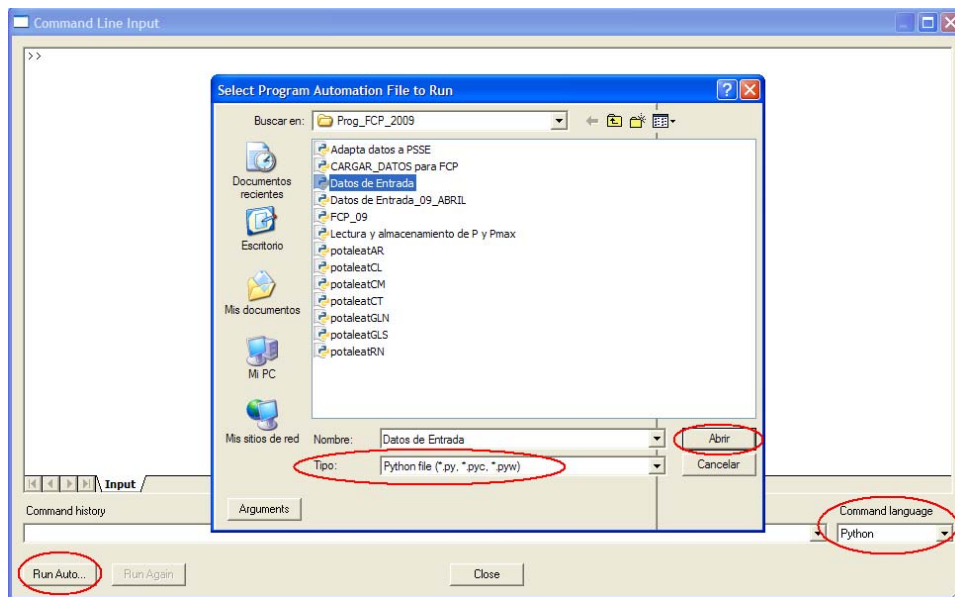


Figura 15. Pantalla PSS/E de “CLI” y “Run Auto”

Al abrirlo se ejecuta automáticamente y de esta forma se almacenan los datos de entrada en PSS/E. Aparecerá en la pantalla el siguiente mensaje:

Los datos de entrada se han cargado correctamente, ejecutar el programa: Lectura y Almacenamiento de P y Pmax

3º Volver a pinchar el botón “Run Auto” y abrir el programa “Lectura y almacenamiento de P y Pmax. py”, se ejecutará y aparecerá en la pantalla el mensaje:

El programa ha finalizado abrir MATLAB y ejecutar: PPMAX fecha_hora

4º Ahora es necesario abrir Matlab y ejecutar el programa: “PPMAX_fecha_hora.m”. Para ello debe buscarse dicho archivo en el directorio donde se haya decidido ubicarlo. En la barra de herramientas superior, figura 16, en “Current Directory” se escribe la ruta y aparecerá el archivo en la ventana “Current Directory”. Con el botón derecho del ratón se pincha sobre el archivo, aparece un desplegable y se elige “Run”. También puede escribirse el nombre del archivo en la ventana de comandos “Command Window”.

De esta forma se consigue que Matlab ejecute este programa y almacene en memoria los valores que contiene, de las potencias de los parques eólicos.

Aparecerá el siguiente mensaje en la ventana de comandos “Command Window”:

Los datos ya se ha cargado, ejecutar: DBD

5º Después se hace exactamente lo mismo para ejecutar el programa “DBD.m”. Aparecerá el siguiente mensaje en la ventana de comandos:

EL PROGRAMA HA FINALIZADO, EJECUTAR EN PSS/E: Adapta datos a PSS/E

En la figura 16 se muestra la secuencia a seguir.

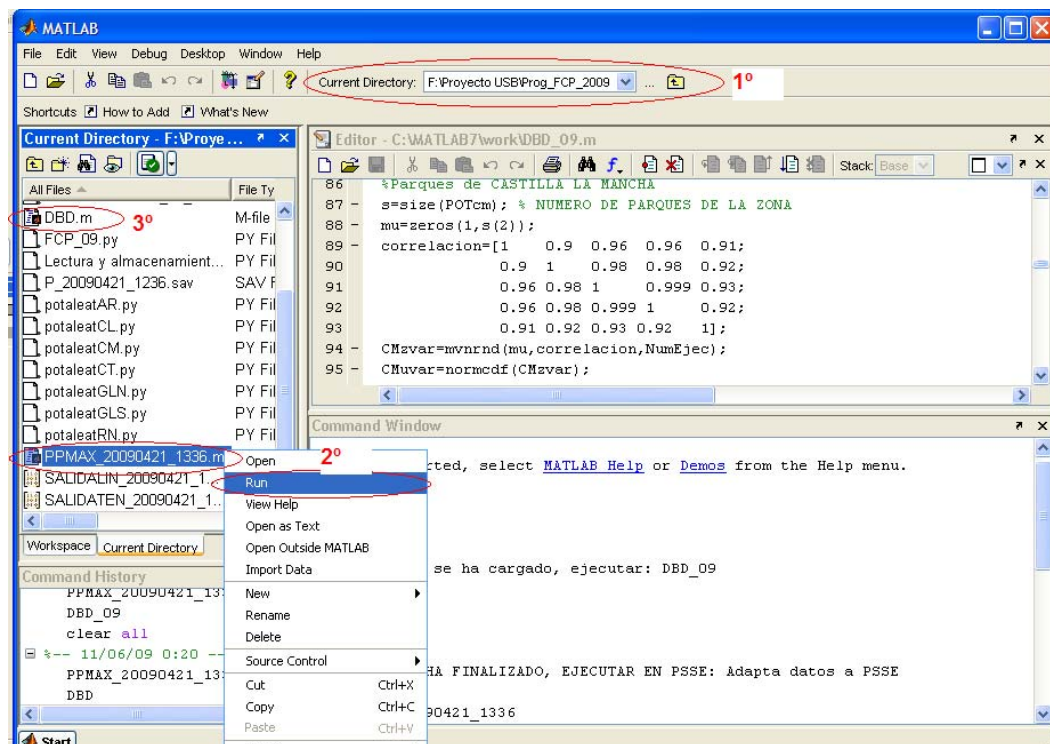


Figura 16. Pantalla de Matlab para ejecución de programas

6º El siguiente paso es ejecutar, en PSS/E, el programa “Adapta datos a PSS/E.py”. Se ejecuta de la misma forma que los anteriores programas de PSS/E, con el botón “CLI”. Al ejecutarse aparecerá el siguiente mensaje en la pantalla:

El programa ha finalizado, ejecute el programa: CARGAR_DATOS para FCP

7º Ejecutar seguidamente en PSS/E el programa “CARGAR_DATOS para FCP.py”. Aparecerá el siguiente mensaje en la pantalla:

Los datos se han cargado, ejecute el programa: FCP_09

8º Finalmente ejecutar en PSS/E, el programa “FCP.py”. Este programa tardará unos minutos, dependiendo del número de ejecuciones que requiera. Al finalizar aparecerá el siguiente mensaje:

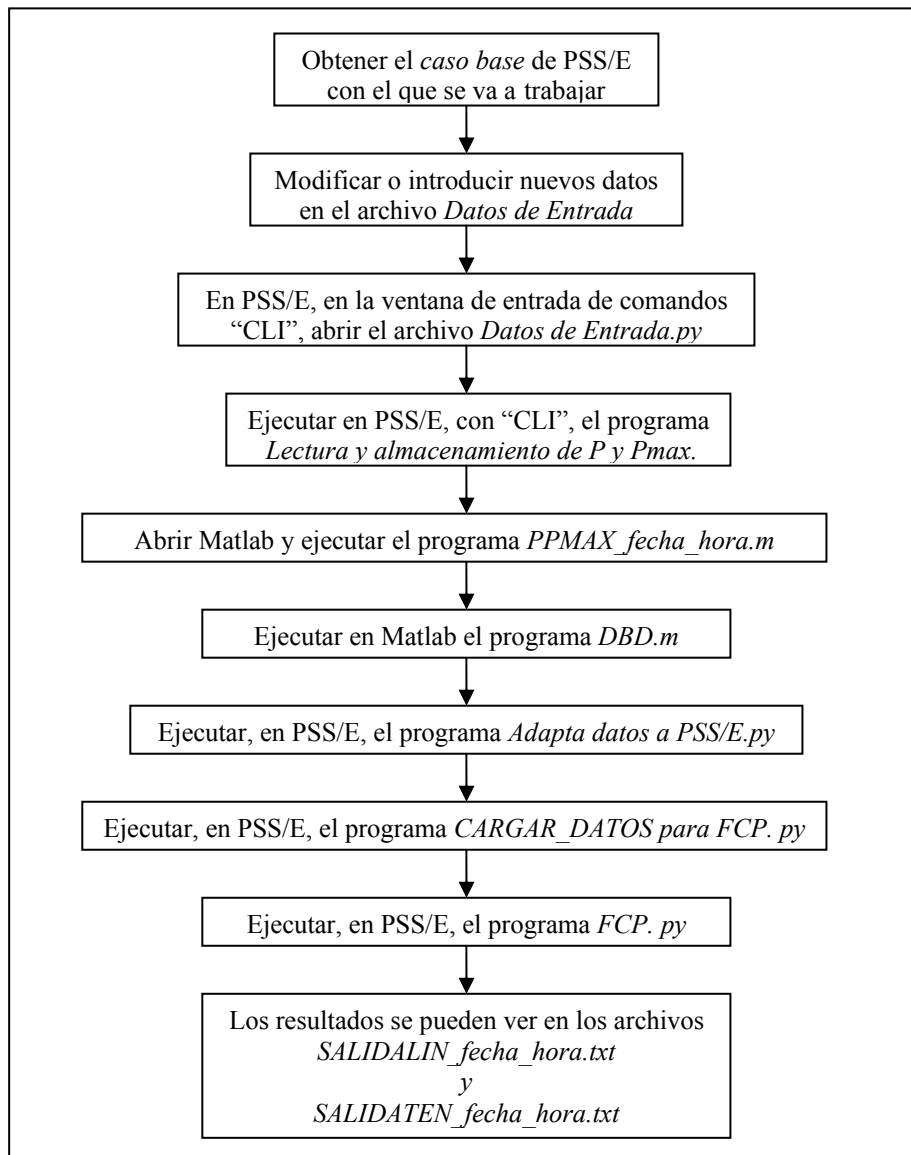
El flujo de cargas probabilístico ha finalizado, los resultados pueden verse en:

F:\Proyecto USB\Prog_FCP_2009\SALIDALIN_20090421_1336.txt
F:\Proyecto USB\Prog_FCP_2009\SALIDATEN_20090421_1336.txt

La ruta y nombre de los archivos de salida será la que se les asignó en el archivo de datos de entrada.

9º Para ver los resultados basta con abrir ambos archivos.

A continuación se muestra un flujograma, a modo de resumen, con los pasos necesarios a seguir para ejecutar un flujo de cargas probabilístico.



Flujograma 2. Pasos necesarios para ejecutar un flujo de cargas probabilístico.

A.2. Programas

A.2.1 Descripción detallada del formato del programa: “Datos de Entrada”.

En él se ha definido:

1º. La ruta en la que se encuentra el **caso base .raw** y su nombre con la extensión, “.raw”.

Ejemplo: `CASOraw=r" F:\Prog_FCP_2009\20090118_08.raw "`

Como se ha indicado antes, debe respetarse el formato de este programa ya que está escrito en código Python; adaptado a PSS/E, para que éste pueda interpretarlo. Por lo tanto sólo se modificará la ruta y el nombre con la extensión, dejando el resto de signos y letras tal y como aparecen en el ejemplo anterior. Esto se aplica al resto de elementos que integran este archivo de datos de entrada.

2º. La ruta en la que se guardará el **caso base .sav** y su nombre con la extensión “.sav”. Este caso será el que, como se indicó en el apartado 7, PSS/E utilizará como caso de partida en cada una de las ejecuciones del programa *FCP*.

Ejemplo: `CASOsav=r" F:\Prog_FCP_2009\20090118_08.sav "`

3º. La ruta y el nombre del archivo de Matlab, con la extensión “.m”, donde se almacenan, agrupadas por zonas geográficas y en forma de vectores, las potencias aleatorias generadas para cada parque eólico de la zona y las potencias máximas de cada parque.

Ejemplo:

`PzonasPMAXzonas="F:\ProyectoUSB\Prog_FCP_2009\PPMAX_20090421_1336.m"`

4º. La ruta y nombres de los archivos de salida, en los que se muestran las circulaciones de las líneas y las tensiones de los nudos. Estos dos archivos tendrán formato de texto con extensión “.txt”, en los cuales aparecerá probabilidad de superar un valor de carga de las líneas en vigilancia y las tensiones de las subestaciones especificadas.

Ejemplo: `SALIDALINEA="F:\Prog_FCP_2009\SALIDALIN_20090118_08.txt"`

`SALIDATENSION="F:\Prog_FCP_2009\SALIDATEN_20090118_08.txt"`

La ruta y nombres de los archivos quedan a elección del usuario, pero lo más lógico y cómodo, para llevar un control de todos los archivos que se generarán durante la ejecución de los distintos programas que conforman el flujo de cargas probabilístico, será nombrar a todos los archivos con la fecha y hora del caso con el que se está trabajando, a parte del nombre particular de cada archivo, si lo tiene.

5º. El número de bus y el identificador de cada generador eólico del sistema a estudiar, separados por zonas geográficas.

En este proyecto como se ha elegido trabajar con el sistema eléctrico español, se han separado los parques eólicos por zonas geográficas, dando por supuesto que los parques eólicos de una misma zona se comportan de manera similar en función de las condiciones meteorológicas que afecten a dicha zona.

Se han definido 7 zonas geográficas que son:

Aragón
Castilla León
Castilla La Mancha
Cataluña
Galicia norte
Galicia sur
Rioja- Navarra

Se ha creado un vector para cada zona y se ha introducido el número de bus y el identificador de cada generador eólico perteneciente a la zona.

También se ha definido un vector, integrado por todos los vectores de las zonas, de forma que la estructura de datos de los parques eólicos es una matriz.

Ejemplo del formato de datos de los parques eólicos:

```
BUSNUGar=[23009,'1',23017,'1',23017,'2',23073,'1',23073,'2',23133,'1',23195,'1',23302,'1',23302,'2',3436,'1']  
BUSNUGcl=[12030,'1',1237,'1',1310,'1',1713,'1',2024,'1',22127,'1',2531,'1',2590,'1',2780,'1',2782,'1']  
BUSNUGcm=[24213,'1',24245,'1',4502,'1',4501,'2',4501,'3']  
BUSNUGct=[3188,'1',3506,'1',3926,'1',3008,'1',3008,'2']  
BUSNUGgln=[1086,'1',1091,'1',1183,'1',1294,'1',21105,'1',41013,'3',1013,'1',1013,'2',1875,'1']  
BUSNUGgls=[1094,'1',1197,'1',1197,'2',21005,'1',21123,'1']  
BUSNUGrn=[22134,'1',2181,'1',22100,'1']  
BUSNUG=[BUSNUGar,BUSNUGcl,BUSNUGcm,BUSNUGct,BUSNUGgln,BUSNUGgls,BUSNUGrn]
```

6º. El número de bus y el identificador de cada carga del sistema a estudiar. Tienen el mismo formato que en el caso de los generadores eólicos.

Ejemplo: `BUSNUC=[22130,'1',22130,'2',1055,'1',2006,'1',2006,'2',2066,'1',2066,'2',2066,'3',22050,'1']`

7º. Los números de bus y el identificador de los generadores térmicos entre los que se repartirá la variación de generación eólica surgida. Tienen el mismo formato que en el caso de los generadores eólicos.

Esta lista de generadores térmicos, se ha configurado atendiendo a los criterios seguidos en la operación del sistema eléctrico español. En dicha lista están los grupos térmicos que, normalmente, participan en la regulación secundaria del sistema, que es la que absorbe en los primeros minutos las variaciones de potencia, ya sea aumento o descenso de la demanda o del resto de la generación del sistema.

Ejemplo: `BUSTERMICA=[1229,'1',1231,'1',1233,'1',1602,'1',1618,'1',1656,'1',1657,'1',1742,'1']`

8º. Los números de buses de las líneas en vigilancia y el identificador de cada una de ellas, de las que queremos conocer las cargas y la probabilidad de sobrepasar un valor de carga determinado, una vez realizado el flujo de cargas probabilístico.

Ejemplo: `BUSLIN=[22245,22130,'1',22050,22185,'1',21075,21245,'1',22335,63142,'1',22134,22245,'1']`

9º. Los números de bus de las subestaciones, de los que queremos conocer la tensión una vez realizado el flujo de cargas probabilístico.

Ejemplo: `BUSTENSION=[11055,21075,22245,23175]`

10º. El número de iteraciones del programa *FCP*, necesario para obtener unos resultados aceptables.

11º. El número de intervalos en los que se desea dividir, el rango en el cual se encuentran los valores calculados de cargas de las líneas y tensiones de las subestaciones, para realizar un análisis en frecuencia de las circulaciones y las tensiones, mostrando los resultados en los archivos de salida, de forma que se observe el número de veces que la circulación de una línea o la tensión de una subestación estará dentro de cada intervalo, así como los límites de dichos intervalos.

12º. El porcentaje de carga de las líneas, del que se quiere conocer la probabilidad de superarlo.

13º. Se ha añadido un mensaje de información al usuario, que aparecerá en la pantalla de PSS/E, para informarle de que los datos de entrada se han cargado correctamente. Este mensaje no debe ser modificado por el usuario.

El aspecto final del programa “Datos de Entrada” será el siguiente:

- Nota: En lenguaje Python todo lo escrito después de la almohadilla (#) lo considera un comentario. En dichos comentarios no admite acentos.

```
# "DATOS DE ENTRADA"

CASOraw=r" F:\Prog_FCP_2009\20090421_1236.raw ""
CASOsav=r" F:\Prog_FCP_2009\20090421_1336.sav ""
PzonasPMAXzonas="C:\MATLAB7\work\PPMAX_20090421_1336.m"
SALIDALINEA="F:\Prog_FCP_2009\SALIDALIN_20090421_1336.txt"
SALIDATENSION="F:\Prog_FCP_2009\SALIDATEN_20090421_1336.txt"

# " Buses de los generadores eolicos separados por zonas"
BUSNUGar=[23009,'1',23017,'1',23017,'2',23073,'1',23073,'2',23133,'1',23195,'1',23302,'1',23302,'2',3436,'1',3666,'1',
3666,'2',3666,'3',3687,'1',3749,'1',3750,'1',3790,'1',3793,'1',3795,'1',3811,'3',3812,'1',3813,'2']
BUSNUGcl=[12030,'1',1237,'1',1310,'1',1713,'1',2024,'1',22127,'1',2531,'1',2590,'1',2780,'1',2782,'1']
BUSNUGcm=[24213,'1',24245,'1',4502,'1',4501,'2',4501,'3']
BUSNUGct=[3188,'1',3506,'1',3926,'1',3008,'1',3008,'2']
BUSNUGln=[1086,'1',1091,'1',1183,'1',1294,'1',21105,'1',41013,'3',1013,'1',1013,'2',1875,'1']
BUSNUGls=[1094,'1',1197,'1',1197,'2',21005,'1',21123,'1']
BUSNUGrn=[22134,'1',2181,'1',22100,'1']
BUSNUG=[BUSNUGar,BUSNUGcl,BUSNUGcm,BUSNUGct,BUSNUGln,BUSNUGls,BUSNUGrn]
```



```

# "Buses de carga"
BUSNUC=[22130,'1',22130,'2',1055,'1',2006,'1',2006,'2',2066,'1',2066,'2',2066,'3',22050,'1',22050,'2',22100,'4',22105,'1',
22105,'2',22105,'5',22130,'1',22130,'2',22130,'3',22165,'1',22165,'2',22135,'1',22135,'2',22200,'1',22200,'2',
22200,'3',22245,'2',22245,'3']

# "Buses de generadores termicos entre los que se repartira la variacion de generacion eolica"
BUSTERMICA=[1229,'1',1231,'1',1233,'1',1602,'1',1618,'1',1656,'1',1657,'1',1742,'1',1743,'1',1744,'1',2480,'1',2485,'1',
2529,'1',3510,'1',3748,'1',4041,'1',4064,'1',4701,'1',4700,'1',4879,'1',4882,'1',4883,'1',4884,'1',5307,'1',5309,'1',
5327,'1',5337,'1',5338,'1',5363,'1',5755,'1',5757,'1',5758,'1',5759,'1',5930,'1',5935,'1',]

# "Lineas de las que queremos analizar las circulaciones"
BUSLIN=[22245,22130,'1',22050,22185,'1',21075,21245,'1',22335,63142,'1',22134,22245,'1'] # el 63142 es Magallon con
ACP abierto

# "Numero de los buses de las subestaciones de las que queremos analizar la tension"
BUSTENSION=[11055,21075,22245,23175] # "Numero de los buses en los que queremos analizar la tension"

# "Numero de iteraciones necesarias para el programa FCP"
NumIter=5

# " Numero de intervalos en los que se desea dividir, el rango en el cual se encuentran los valores calculados de cargas
# "de las lineas y tensiones de las subestaciones, para realizar un analisis en frecuencia."
numedintervalos=10

# "Porcentaje de carga de las lineas, del que se quiere conocer la probabilidad de superarlo"
UMBRAL=90

print "Los datos de entrada se han cargado correctamente, ejecutar el programa: Lectura y Almacenamiento de P y Pmax"

```

A.2.2 Descripción detallada del código del programa: “Lectura y Almacenamiento de P y Pmax” .

Seguidamente se detallan las instrucciones utilizadas y los pasos seguidos para la creación del programa.

1º. Lectura y convergencia del “caso base.raw”. Se lee el caso especificado en el archivo de datos de entrada, cuya ruta y nombre ha sido almacenado en la variable “CASOraw”. También se ha especificado que la versión del caso a leer es la 29. La convergencia del caso se realiza con un flujo de cargas en alterna utilizando el método de Newton-Raphson.

Las instrucciones utilizadas para la lectura y convergencia son:

```
psspy.readrawversion(0,r""29"",CASOraw)
```

```
psspy.fdns([0,0,0,1,1,0,99,0])
```

2º. Almacenamiento, como “caso base.sav”, del “caso base.raw”, leído y convergido anteriormente. El caso se almacena en la ruta y con el nombre que se especificó en el archivo de datos de entrada, que se encuentran en la variable “CASOsav”.

La instrucción es:

```
psspy.save(CASOsav)
```

3º. Se han creado 7 vectores vacíos; uno para cada zona geográfica en las que se ha dividido España, en los que se almacenarán las potencias previstas por los parques eólicos de cada zona y otros 7 vectores vacíos, en los que se almacenarán las potencias máximas de cada parque.

Ejemplo:

Par=[]	; PMAxar=[]	# ARAGON
Pcl=[]	; PMAxcl=[]	# CASTILLA LEON
Pcm=[]	; PMAxcm=[]	# CASTILLA LA MANCHA
Pct=[]	; PMAxct=[]	# CATALUNA
Pgln=[]	; PMAxgln=[]	# GALICIA NORTE
Pgls=[]	; PMAxgls=[]	# GALICIA SUR
Prn=[]	; PMAxrn=[]	#RIOJA-NAVARRA

También se han creado dos vectores (Pzonas y PMAxzonas), compuestos por los vectores de potencias previstas de cada zona y con los de potencias máximas de los parques, de cada zona. De esta forma los datos están estructurados en forma matricial y es más fácil trabajar con ellos.

Ejemplo:

Pzonas=[Par,Pcl,Pcm,Pct,Pgln,Pgls,Prn]
PMAxzonas=[PMAxar,PMAxcl,PMAxcm,PMAxct,PMAxgln,PMAxgls,PMAxrn]

4º. La lectura y almacenamiento de las potencias previstas para los parques eólicos de cada zona y las potencias máximas de los mismos, se realiza mediante un bucle de programación en el que se comienza inicializando a cero, tanto la variable que contará el número de zonas “z” como la que contará el número de parques de cada zona “ng”, esta se inicializa a cero al cambiar de zona geográfica.

Lo siguiente es leer del caso cargado en PSS/E: “caso base.sav”. Para cada zona, la potencia prevista y máxima de cada parque; para esto es necesario utilizar el número de bus del parque y su identificador, estos datos se encuentran almacenados en la matriz “BUSNUG”, definida en el archivo de datos de entrada.

A la hora de realizar esta lectura; primero se comprueba si el parque eólico está conectado a la red y si la potencia generada es positiva. Si alguna de estas condiciones no se cumple, se iguala a cero la potencia generada por el parque; ya que se ha observado en los casos con los que se ha trabajado, que algunos valores de potencia generada por los parques son negativos (generalmente valores pequeños). Esto se debe fundamentalmente a errores de medida causados posiblemente por los convertidores de medida de los parques eólicos, ya que el error de los convertidores con valores de potencia cercanos a cero hace que las medidas no sean fiables y en algunos casos marquen valores negativos. Estos valores son recibidos por Red Eléctrica de España en su sistema de control y al generarse el “caso base .raw”, quedan reflejados en él.

También se comprueba si la potencia prevista para cada parque es menor que su potencia máxima que figura en el caso. Si no es así se aumenta la potencia máxima 0,5 MW por encima del valor de potencia previsto; ya que si no, no se podrían realizar los cálculos posteriores.

Se ha elegido aumentar 0,5 MW por no tener el dato de la potencia máxima real, y suponer que el parque eólico en cuestión está generando, prácticamente, la máxima potencia. La diferencia de 0,5 MW respecto a la potencia prevista es despreciable y no introduce errores en el cálculo de las potencias aleatorias generadas por el parque.

Una potencia máxima menor que la potencia prevista se ha comprobado que puede ocurrir, ya que al ser la potencia máxima de cada parque, un dato que introduce Red Eléctrica de España en su modelo de red eléctrica español y por tanto en los casos de PSS/E con los que se ha trabajado en este proyecto, puede que se instalen nuevos parques eólicos en un nudo de la red o se amplíen los existentes y los datos de su potencia máxima no se actualicen en el mismo momento, en que son puestos en servicio.

Una vez leídas las potencias previstas y máximas de los parques, éstas se almacenan en los vectores creados. De manera que se recorren los parques eólicos de una zona leyendo y almacenando sus potencias, se pasa a la siguiente zona y se vuelve a realizar la lectura y almacenamiento de potencias.

El bucle de programación acaba cuando se llega al último parque eólico de la última zona.

En ese momento los vectores de potencias previstas y máximas contendrán todos los valores necesarios para que Matlab pueda generar las potencias aleatorias para cada parque eólico.

Aquí se muestra el bucle de programación para realizar lo explicado anteriormente:

```
z=0 # "Contador de las zonas geograficas, donde estan los PE, en las que se ha dividido Espana
while z<len(BUSNUG):
    ng=0 # "Contador del numero de generadores eolicos, en cada zona"
    while ng<len(BUSNUG[z]):
        # "De cada nudo lo primero es ver si esta conectado y si la potencia generada es positiva"
        st=psspy.macint(BUSNUG[z][ng],BUSNUG[z][ng+1], 'STATUS')
        status=st[1] # "Ponemos st[1] porque st es (0,estatus)"
        Pg=psspy.macdat(BUSNUG[z][ng],BUSNUG[z][ng+1], 'P')
        P=Pg[1] # "Ponemos Pg[1] porque Pg es (0,potencia generada)"
        Pgmax=psspy.macdat(BUSNUG[z][ng],BUSNUG[z][ng+1], 'PMAX')
        PMAX=Pgmax[1] # "Ponemos Pgmax[1] porque Pgmax es (0,potencia maxima)"

        if status==0 or P<0:
            P=0
        if P>=PMAX:
            PMAX=P+0.5

        Pzonas[z].append(P)
        PMAXzonas[z].append(PMAX)
        ng=ng+2
    z=z+1
```

5º. Por último para generar el archivo de texto “.m”, con los datos de las potencias, que el programa “DBD” de Matlab utilizará para calcular las potencias aleatorias generadas por cada parque; se han creado dos nuevos vectores compuestos por los nombres de los vectores de potencias previstas y máximas, necesarios dichos nombres para componer dicho archivo.

Para escribir los datos en el archivo “PPMAX_fecha_hora.m”, cuya ruta y nombre definió el usuario en el archivo de datos de entrada y se almacenó en la variable “PzonasPMAXzonas”, se comienza abriendo el archivo. Lo primero que se ha escrito es la instrucción de Matlab “clear all” para eliminar todas las variables creadas en Matlab, ya que se ha observado que en ocasiones se producen errores de cálculo por almacenar variables antiguas. De esta manera

cada vez que se carguen los datos de las potencias previstas y máximas contenidas en este archivo se realiza un borrado de todas las variables anteriores.

Lo siguiente que se ha hecho, mediante un bucle de programación, es introducir en dicho archivo los valores de potencias previstas y máximas, en forma de vectores. Para esto ha sido necesario crear los vectores de forma que Matlab pueda leerlos.

Por último se escribe el número de iteraciones necesarias para el cálculo del flujo de cargas probabilístico, dato procedente del archivo de datos de entrada y dos avisos para informar al usuario que este programa ha finalizado correctamente y otro que aparecerá al cargar el archivo creado, en Matlab, para informarle de que los datos se han cargado correctamente.

Se muestra el código de programación que ha resultado al programar este último punto.

```
# Se crean dos vectores con os nombres de los vectores que contendran los datos de las potencias, en el archivo .m
POTzonas=['POTar','POTcl','POTcm','POTct','POTgln','POTgls','POTrn']
POTMAXzonas=['POTMAXar','POTMAXcl','POTMAXcm','POTMAXct','POTMAXgln','POTMAXgls','POTMAXrn']

# "Abrimos y Escribimos las potencias previstas y las potencias maximas de cada parque, por zonas"

d=open(PzonasPMAZzonas,"w")
d.write(str('clear all'))
z=0
while z<len(BUSNUG):
    d.write("\n"+str(POTzonas[z])+"="+str(Pzonas[z])+";\n"+str(POTMAXzonas[z])+"="+str(PMAZzonas[z])+";\n")
    z=z+1
d.write("\n NumIter="+str(NumIter)+";\n"+"Los datos ya se ha cargado, ejecutar: DBD_09")
d.close()

print "\n\n El programa ha finalizado abrir MATLAB y ejecutar: PPMAX_fecha_hora"
```

Seguidamente se muestra el programa completo: “*Lectura y Almacenamiento de P y Pmax*” y por último el archivo: “PPMAX_fecha_hora.m” que se ha generado con este programa y contiene la potencias eólicas previstas y máximas de los parques eólicos, que el programa “DBD” de Matlab utilizará como datos de partida.

“*Lectura y Almacenamiento de P y Pmax*”

```
# "Leemos el caso .sav "
psspy.case(CASOsav) # "Se lee el caso"

# "Convergemos el caso .raw "
psspy.fdns([0,0,0,1,1,0,99,0])
# "Guardamos el caso convergido como .sav ,que sera el que utilizara la segunda parte del programa en cada ejecucion"
psspy.save(CASOsav)

# "Creamos los vectores para cada zona, donde almacenaremos las potencias previstas y maximas, de cada parque"
Par=[]; PMAZar=[] # ARAGON
Pcl=[]; PMAZcl=[] # CASTILLA LEON
Pcm=[]; PMAZcm=[] # CASTILLA LA MANCHA
Pct=[]; PMAZct=[] # CATALUNA
Pgln=[]; PMAZgln=[] # GALICIA NORTE
Pgls=[]; PMAZgls=[] # GALICIA SUR
Prn=[]; PMAZrn=[] #RIOJA-NAVARRA
Pzonas=[Par,Pcl,Pcm,Pct,Pgln,Pgls,Prn]
PMAZzonas=[PMAZar,PMAZcl,PMAZcm,PMAZct,PMAZgln,PMAZgls,PMAZrn]
```

```

z=0 # "Contador de las zonas, donde estan los PE, en las que se ha dividido Espana
while z<len(BUSNUG):
    ng=0 # "Contador del numero de generadores eolicos, en cada zona"
    while ng<len(BUSNUG[z]):
        # "De cada nudo lo primero es ver si esta conectado y si la potencia generada es positiva"
        st=psspy.macint(BUSNUG[z][ng],BUSNUG[z][ng+1], 'STATUS')
        status=st[1] # "Ponemos st[1] porque st es (0,estatus)"
        Pg=psspy.macdat(BUSNUG[z][ng],BUSNUG[z][ng+1], 'P')
        P=Pg[1] # "Ponemos Pg[1] porque Pg es (0,potencia generada)"
        Pgmax=psspy.macdat(BUSNUG[z][ng],BUSNUG[z][ng+1], 'PMAX')
        PMAX=Pgmax[1] # "Ponemos Pgmax[1] porque Pgmax es (0,potencia maxima)"

        if status==0 or P<0:
            P=0
        if P>=PMAX:
            PMAX=P+0.5

        Pzonas[z].append(P)
        PMAXzonas[z].append(PMAX)
        ng=ng+2
    z=z+1

# Se crean dos vectores con os nombres de los vectores que contendran los datos de las potencias, en el archivo .m
POTzonas=['POTar','POTcl','POTcm','POTct','POTgln','POTgls','POTrm']
POTMAXzonas=['POTMAXar','POTMAXcl','POTMAXcm','POTMAXct','POTMAXgln','POTMAXgls','POTMAXrm']

# "Abrimos y Escribimos las potencias previstas y las potencias maximas de cada parque, por zonas"

d=open(PzonasPMAXzonas,"w")
d.write(str('clear all'))
z=0
while z<len(BUSNUG):
    d.write("\n"+str(POTzonas[z])+"="+str(Pzonas[z])+";\n"+str(POTMAXzonas[z])+"="+str(PMAXzonas[z])+";\n")
    z=z+1
d.write("\n NumIter="+str(NumIter)+";\n"+"Los datos ya se ha cargado, ejecutar: DBD_09'")
d.close()

print "\n\n El programa ha finalizado abrir MATLAB y ejecutar: PPMAX_fecha_hora"

```

“PPMAX_fecha_hora.m”

```

clear all
POTar=[0.0, 0, 0, 0, 0.81999999284744263, 1.1599999666213989, 0, 3.809999942779541, 3.809999942779541, 0.0,
12.610000610351563, 10.609999656677246, 13.609999656677246, 0.0, 0.0, 5.429999828338623, 0, 0, 0.0, 0.0, 0.0];
POTMAXar=[50.0, 50.0, 50.0, 48.0, 48.0, 50.0, 50.0, 57.0, 57.0, 90.0, 30.000001907348633, 30.000001907348633,
30.000001907348633, 40.0, 100.0, 40.0, 0.5, 0.5, 0.5, 40.0, 40.0, 40.0];

POTcl=[7.0700006484985352, 0.0, 9.3400001525878906, 0, 19.579999923706055, 0, 0, 28.470001220703125, 0];
POTMAXcl=[90.0, 40.0, 40.0, 0.5, 50.0, 88.0, 0.5, 50.0, 0.5];

POTcm=[0, 0, 126.12001037597656, 0, 10.0];
POTMAXcm=[86.0, 200.0, 430.00003051757813, 0.5, 38.0];

POTct=[0.0, 0, 3.5099997520446777];
POTMAXct=[28.0, 0.5, 75.0];

POTgln=[0, 0, 31.569999694824219, 0, 8.4499998092651367, 0, 0, 0];
POTMAXgln=[65.0, 81.0, 130.0, 70.0, 31.0, 0.5, 0.5, 0.5];

POTgls=[17.989999771118164, 5.0, 5.0, 77.639999389648438, 5.0900001525878906];
POTMAXgls=[60.000003814697266, 6.0, 6.0, 128.0, 49.0];

POTrm=[21.520000457763672, 7.9700002670288086, 0.0];
POTMAXrm=[100.0, 185.0, 110.0];

NumEjec=13;
'Los datos ya se ha cargado, ejecutar DBD'

```

A.2.3 Descripción detallada del código del programa: “DBD”.

Se ha estructurado el programa de manera que se calculan zona a zona las potencias aleatorias generadas por los parques de cada una de ellas. Por tanto se va a describir el código empleado para los cálculos de una zona y el resto del código será idéntico, solo cambiarán los nombres de las variables.

Se ha seguido el criterio de nombrar a las variables precedidas o seguidas de las iniciales de la zona en mayúsculas.

Las iniciales de cada zona utilizadas son:

AR → Aragón
CL → Castilla León
CM → Castilla la Mancha
CT → Cataluña
GLN → Galicia Norte
GLS → Galicia Sur
RN → Rioja-Navarra

Las variables utilizadas son, p.e. para Castilla la Mancha:

CMzvar → Matriz de números aleatorios, generados mediante una FDP normal

CMuvar → Matriz de probabilidad de ocurrencia de los números aleatorios generados anteriormente.

potalpuCM → Matriz de potencias aleatorias en p.u, generadas mediante una FDP beta

potaleatCM → Matriz de potencias aleatorias en MW.

- Descripción del código:

1º. Comienza el cálculo obteniendo el número de parques de la zona. Mediante la instrucción siguiente, se contabilizan el número de elementos que posee el vector de potencias previstas por los parques de la zona, procedentes de la lectura del caso en PSS/E:

```
s=size(POTcm)
```

2º. Se crea un vector de ceros, tantos como parques eólicos tiene la zona. Este vector contiene los valores de la media (μ) de una distribución normal multivariable:

```
mu=zeros(1,s(2))
```

3º. Se introduce la matriz de correlación para los parques eólicos de la zona. Esta matriz es simétrica semidefinida positiva, de forma que posee unos en la diagonal principal y el resto de elementos son los coeficientes de correlación de cada parque con el resto de parques de la zona. Tendrá tantas filas y columnas como parques haya en la zona.

Por ejemplo:

El elemento (1,1) de la matriz es 1 por ser la correlación de la potencia generada por primer parque con sí misma.

El elemento (1,2) de la matriz es 0.9 p.e, porque entre la potencia generada por el primer parque y por el segundo existe dicha correlación. El elemento (2,1) será igualmente 0.9, correlación entre la potencia del segundo parque y del primero.

Ejemplo de matriz de correlación:

```
correlacion=[ 1    0.9  0.96  0.96  0.91
              0.9  1    0.98  0.98  0.92
              0.96 0.98 1    0.999 0.93
              0.96 0.98 0.999 1    0.92
              0.91 0.92 0.93  0.92  1   ]
```

4º. Con la media (μ), la matriz de correlación y el número de iteraciones del flujo de cargas probabilístico (dato de entrada) se genera una matriz de números aleatorios, en torno a la media de la distribución (0) y considerando los coeficientes de correlación, según una distribución normal multivariable. Dicha matriz es necesaria para el siguiente paso y tendrá tantas columnas como parques posea la zona y tantas filas como número de iteraciones:

`CMzvar=mvnrnd(mu,correlacion,NumIter)`

Se muestra un ejemplo de matriz de números aleatorios generada con esta instrucción:

```
CMzvar =    0.4694  0.6634  0.6864  0.6214  0.6917
           -0.9036 -0.9020 -1.0592 -1.0286 -0.7597
           0.0359 -0.8632 -0.5191 -0.5368 -0.1642
           -0.6275 -0.5070 -0.6265 -0.6163 -0.0252
           0.5354  1.1762  0.8487  0.8446  0.6958
```

5º. Se calcula la FDP normal acumulativa de cada número aleatorio de la matriz obtenida anteriormente, obteniéndose la probabilidad de ocurrencia de cada uno de estos valores. El resultado se obtiene en forma de matricial:

`CMuvar=normcdf(CMzvar)`

Ejemplo de matriz de probabilidades:

```
CMuvar =    0.6806  0.7465  0.7538  0.7328  0.7554
           0.1831  0.1835  0.1448  0.1518  0.2237
           0.5143  0.1940  0.3018  0.2957  0.4348
           0.2652  0.3061  0.2655  0.2689  0.4900
           0.7038  0.8802  0.8020  0.8008  0.7567
```

6º. Lo siguiente que se necesita conocer son los valores de los parámetros “a” y “b” de la FDP beta; con ellos y la matriz de probabilidades anterior se obtienen las potencias aleatorias generadas por cada parque en base a dicha FDP beta dependiente.

Para esto se ha utilizado un bucle de programación que recorre cada parque de la zona calculando dichos parámetros y las potencias aleatorias generadas por el parque (una para cada ejecución del flujo de cargas probabilístico).

Se inicia el cálculo de los parámetros “a” y “b” de un parque, obteniendo la potencia media como potencia prevista entre potencia máxima. La potencia prevista y máxima de cada parque se encuentran almacenadas en los vectores procedentes de PSS/E, como se indicó anteriormente.

Introduciendo la potencia media obtenida, en la fórmula empírica de la desviación típica, se obtiene ésta.

$$\sigma = -0.79257 \cdot (P_{prev})^2 + 0.77991 \cdot P_{prev} + 0.042078$$

A partir de la desviación típica se obtiene la varianza de las potencias del parque y por último con la potencia media calculada y la varianza se calculan los parámetros “a” y “b” de la distribución beta.

Una vez obtenidos estos parámetros, se calculan las potencias aleatorias generadas por cada parque, en p.u., mediante la inversa de la FDP beta dependiente, ya que se parte de los valores de probabilidad anteriormente generados. Como se puede observar en la siguiente gráfica de una distribución beta, figura 22, en el eje Y se encuentran los valores de probabilidad y lo que aquí se busca son los valores del eje X, en nuestro caso potencia en p.u, que se corresponden con la probabilidad dada y los parámetros “a” y “b”. [1]

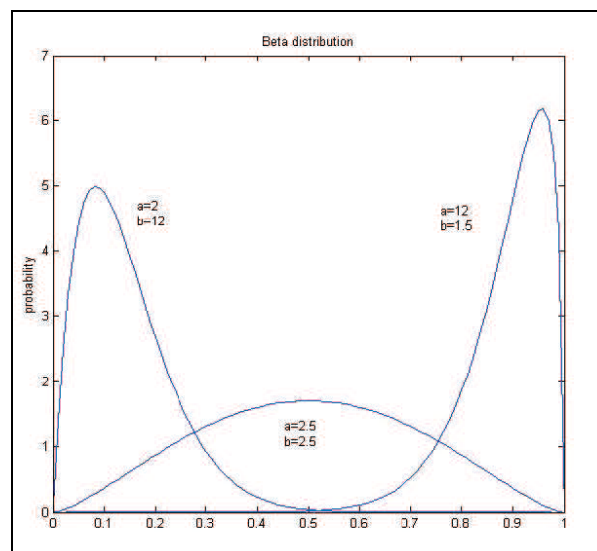


Figura 22. Distribución beta.

Por tanto en cada ejecución del bucle de programación se calcula, para cada parque, tantas potencias aleatorias como numero de iteraciones del flujo de cargas sean necesarias, mediante la instrucción:

$$potalpuCM = \text{betainv}(\text{CMuvar}(:,n), a, b)$$

Esta instrucción necesita los valores de probabilidad para el parque, que se encuentran almacenados por columnas en la matriz de probabilidades de los parques de la zona.

El resultado de todo esto es un vector con los valores en p.u. de las potencias aleatorias generadas por el parque. Estas potencias en p.u. es necesario transformarlas a potencia en MW, esto se consigue multiplicando cada uno de los valores por la potencia máxima del parque.

Al final de cada ejecución del bucle, se almacenan las potencias generadas aleatoriamente, en MW; en una matriz (*potaleatZONA*) donde se encuentran, por filas, todas las potencias generadas para cada parque de la zona. Esta matriz tendrá tantas columnas como número de iteraciones sean necesarias.

A continuación se muestra el bucle de programación que realiza los cálculos anteriormente mencionados y un ejemplo de la matriz de potencias generadas, *potaleatZONA*:

```
for n=1:s(2)
    potmed=POTcm(n)/POTMAXcm(n);
    desvtip= -(0.79257*potmed^2)+ 0.77991*potmed + 0.042078;
    varianza=desvtip^2;
    a=potmed*((potmed*(1-potmed)/varianza)-1);
    b=(1-potmed)*((potmed*(1-potmed)/varianza)-1);
    potalpuCM=betainv(CMuvar(:,n),a,b);
    potaleatCM(n,:)=potalpuCM*POTMAXcm(n);
end
```

```
potaleatRN = 97.4566  85.0485  60.4704  88.6021  89.3754
             182.8402 169.9404 145.2204 168.3039 173.7934
             76.6466  38.2273  20.6814  38.9281  42.3502
```

7º. El penúltimo paso para obtener un archivo de texto en el que se encuentren en forma matricial las potencias generadas por los parques eólicos de una zona, es preparar la matriz de potencias mostrada anteriormente para que PSS/E y Python puedan leerla e interpretarla. Esta matriz de potencias puede contener elementos con el valor “NaN” que es el resultado de que un parque no estuviese generando potencia en el momento en el que se generó el caso de PSS/E con el que se está trabajando. Si el parque no estaba generando potencia, al realizar los cálculos de los parámetros “a” y “b”, estos resultan cero y no se pueden obtener valores aleatorios de potencia generada mediante la distribución beta, por lo que Matlab deja vacío el elemento correspondiente de la matriz de potencias generadas (*potaleatZONA*) y al mostrar dicha matriz aparece “NaN” (not a number).

Mediante un bucle de programación y con la instrucción de Matlab “find(isnan....)” se recorre la matriz (*potaleatZONA*) y si algún elemento aparece como “NaN” se sustituye por cero.

El código creado es el siguiente:

```
for f=1:s(2)
    for c=1:NumIter
        e=find(isnan(potaleatCM(f,c)));
        if e==1
            potaleatCM(f,c)=0;
        end
    end
end
```

8°. El último paso será crear un archivo de texto, en el que se almacenará la matriz de potencias generadas (potaleatZONA).

Este archivo tiene extensión “.py” (python), pero se podría haber elegido la extensión “.txt”, porque lo único que se hará con estos archivos es leerlos en PSS/E para extraer su información. Para ello es necesario indicar la ruta donde se guardará el archivo, su nombre con extensión y lo que se quiere escribir en él, en este caso la matriz (potaleatZONA).

Se muestra el código creado:

```
csvwrite('F:\Proyecto USB\Prog_FCP_2009\potaleatCM.py',potaleatCM)
```

Como se dijo al inicio de este apartado todo lo descrito hasta aquí son las instrucciones necesarias para realizar los cálculos de potencias aleatorias de los parques de una zona, el programa *DBD* realiza estos cálculos para las 7 zonas en las que se ha dividido la generación eólica en España.

Las modificaciones del código, para las diferentes zonas son los nombres de las variables.

Seguidamente se muestra el código del programa para una de las zonas (Cataluña):

```
%Parques de CATALUÑA

s=size(POTct); % NUMERO DE PARQUES DE LA ZONA
mu=zeros(1,s(2));
correlacion=[ 1  0.8  0.5  0.6  0.6;
              0.8  1  0.8  0.6  0.6;
              0.5  0.8  1  0.75 0.75;
              0.6  0.6 0.75 1  0.9;
              0.6  0.6 0.75 0.9  1 ];

CTzvar=mvnrnd(mu,correlacion,NumIter);
CTuvar=normcdf(CTzvar);

for n=1:s(2)
    potmed=POTct(n)/POTMAXct(n);
    desvtip= -(0.79257*potmed^2)+ 0.77991*potmed + 0.042078;
    varianza=desvtip^2;
    a=potmed*((potmed*(1-potmed)/varianza)-1);
    b=(1-potmed)*((potmed*(1-potmed)/varianza)-1);
    potalpuCT=betainv(CTuvar(:,n),a,b);
    potaleatCT(n,:)=potalpuCT*POTMAXct(n);
end

for f=1:s(2) %Los elementos de la matriz que no contienen valor se sustituyen por cero.
    for c=1:NumIter
        e=find(isnan(potalpuCT(f,c)));
        if e==1
            potaleatCT(f,c)=0;
        end
    end
end

csvwrite('F:\Proyecto USB\Prog_FCP_2009\potaleatCT.py',potaleatCT)
```

A.2.4 Descripción detallada del código del programa “Adapta datos a PSS/E”.

A continuación se describe el código del programa “Adapta datos a PSS/E”, creado en lenguaje Python, para transformar los datos de potencias de los parques, procedentes de Matlab.

En Python no se pueden definir matrices como tal, por lo que la forma de crearlas es mediante un anidamiento de vectores. Primero se crea el vector que dará el número de columnas a la matriz, en este caso tendrá tantos elementos como número de iteraciones se requieran para el flujo de cargas probabilístico y después ese vector se copia tantas veces como número de filas se desee en el matriz, en este caso tantas como parques eólicos haya cada zona.

El programa comienza creando el archivo de datos donde se almacenarán las potencias de los parques de las 7 zonas. Este archivo debe tener extensión “.py” ya que será necesario que PSS/E con lenguaje Python lo lea. Es necesario especificar la ruta y el nombre con la extensión del archivo.

Seguidamente se abre dicho archivo creado, en modo escritura (“w”).

Se muestra el código:

```
POTMATLAB="F:\Proyecto USB\Prog_FCP_2009\CARGAR_DATOS.py"
```

```
dm=open(POTMATLAB,"w")
```

Una vez creado y abierto este archivo es necesario introducirle los datos de los 7 archivos procedentes de Matlab. Para ello se irán leyendo cada uno de estos archivos y escribiendo los datos en este archivo de datos comunes, en el formato adecuado.

Se describe el código creado para realizar lo detallado anteriormente, se muestra para una sola zona.

1º. Lo primero será abrir el archivo procedente de Matlab con las potencias aleatorias generadas por los parques de la zona “potaleatZONA.py”. Se especifica la ruta en la que se encuentra el archivo y el modo en el que se abre, modo lectura (‘r’).

```
psspy.pp=open('F:\Proyecto USB\Prog_FCP_2009\potaleatAR.py','r')
```

2º. Lo siguiente será escribir el nombre de la matriz que contendrá las potencias aleatorias generadas, seguido de un signo igual y un corchete, esto es necesario para que quede una matriz con el formato con el que trabaja Python.

```
dm.write(str(' PPar = [ '))
```

3º. Después mediante un bucle se copia cada línea de texto del archivo procedente de Matlab “potaleatZONA.py”. Cada línea de este archivo es una fila de la matriz de potencias aleatorias generadas por los parques de la zona. En cada fila de esta matriz se encuentran, como se dijo anteriormente, tantas potencias como número de iteraciones del flujo de cargas probabilístico sean necesarias

Es necesario escribir un corchete al inicio y otro al final de cada fila de datos, seguido de una coma (formato matricial en Python)

```
pe=0
while pe<len(BUSNUGar)/2:
    pp=psspy.pp.readline()
    dm.write( '[' + str(pp) + '], ' )
    pe=pe+1
```

4º. Una vez copiadas todas las filas de la matriz es necesario cerrarla con un corchete; se deja una línea en blanco de separación con la siguiente matriz de la próxima zona.

El proceso se repite para todas las zonas de generación eólica y al final de este archivo creado, “CARGAR_DATOS para FCP”, se escribe un mensaje de información al usuario, para que al leer PSS/E dicho archivo, el usuario pueda saber si se han cargado los datos correctamente.

Después de esto es necesario cerrar el archivo creado y se escribe un mensaje que aparecerá en la pantalla de PSS/E, indicando al usuario que el programa ha finalizado con éxito y el próximo programa a ejecutar en PSS/E.

A continuación se muestra el código completo del programa “*Adapta datos a PSS/E*” y el archivo resultante “*CARGAR_DATOS para FCP*” que será el siguiente programa a ejecutar en PSS/E.

- Código del programa: “*Adapta datos a PSS/E*”

```
# Programa que transforma el formato de los archivos con las potencias aleatorias de los parques eolicos, calculadas en
# Matlab, en un formato que python pueda leer, crea una matriz para cada zona, PPzona, con dichas potencias.

POTMATLAB="F:\Proyecto USB\Prog_FCP_2009\CARGAR_DATOS.py"
dm=open(POTMATLAB,"w")

# "Creacion de la matriz con las potencias, calculadas en Matlab, de los parques de Aragon"
psspy.pp=open('F:\Proyecto USB\Prog_FCP_2009\potaleatAR.py','r') # los archivos potaleatzona.py pueden ser de texto .txt
dm.write(str('PPar=[]'))
pe=0
while pe<len(BUSNUGar)/2:
    pp=psspy.pp.readline()
    dm.write('['+str(pp)+'],')
    pe=pe+1
dm.write(']\n\n')

# "Creacion de la matriz con las potencias, calculadas en Matlab, de los parques de Castilla Leon"
psspy.pp=open('F:\Proyecto USB\Prog_FCP_2009\potaleatCL.py','r') # los archivos potaleatzona.py pueden ser de texto .txt
dm.write(str('PPcl=[]'))
pe=0
while pe<len(BUSNUGcl)/2:
    pp=psspy.pp.readline()
    dm.write('['+str(pp)+'],')
    pe=pe+1
dm.write(']\n\n')

# "Creacion de la matriz con las potencias, calculadas en Matlab, de los parques de Castilla La Mancha"
psspy.pp=open('F:\Proyecto USB\Prog_FCP_2009\potaleatCM.py','r') # los archivos potaleatzona.py pueden ser de texto .txt
dm.write(str('PPcm=[]'))
pe=0
while pe<len(BUSNUGcm)/2:
    pp=psspy.pp.readline()
```

```

    dm.write([''+str(pp)+''],')
    pe=pe+1
dm.write('']\n\n')

# "Creacion de la matriz con las potencias, calculadas en Matlab, de los parques de Cataluna"
psspy.pp=open('F:\Proyecto USB\Prog_FCP_2009\potaleatCT.py','r') # los archivos potaleatzona.py pueden ser de texto .txt
dm.write(str('PPct=[]'))
pe=0
while pe<len(BUSNUGct)/2:
    pp=psspy.pp.readline()
    dm.write([''+str(pp)+''],')
    pe=pe+1
dm.write('']\n\n')

# "Creacion de la matriz con las potencias, calculadas en Matlab, de los parques de Galicia norte"
psspy.pp=open('F:\Proyecto USB\Prog_FCP_2009\potaleatGLN.py','r') # los archivos potaleatzona.py pueden ser de texto .txt
dm.write(str('PPgln=[]'))
pe=0
while pe<len(BUSNUGgln)/2:
    pp=psspy.pp.readline()
    dm.write([''+str(pp)+''],')
    pe=pe+1
dm.write('']\n\n')

# "Creacion de la matriz con las potencias, calculadas en Matlab, de los parques de Galicia sur"
psspy.pp=open('F:\Proyecto USB\Prog_FCP_2009\potaleatGLS.py','r') # los archivos potaleatzona.py pueden ser de texto .txt
dm.write(str('PPgls=[]'))
pe=0
while pe<len(BUSNUGgls)/2:
    pp=psspy.pp.readline()
    dm.write([''+str(pp)+''],')
    pe=pe+1
dm.write('']\n\n')

# "Creacion de la matriz con las potencias, calculadas en Matlab, de los parques de Rioja y Navarra"
psspy.pp=open('F:\Proyecto USB\Prog_FCP_2009\potaleatRN.py','r') # los archivos potaleatzona.py pueden ser de texto .txt
dm.write(str('PPrn=[]'))
pe=0
while pe<len(BUSNUGrn)/2:
    pp=psspy.pp.readline()
    dm.write([''+str(pp)+''],')
    pe=pe+1
dm.write('']\n\n')

dm.write(str('print')+""+str('Los datos se han cargado, ejecute el programa: FCP_09')+""')

dm.close()

print 'El programa ha finalizado, ejecute el programa: CARGAR_DATOS para FCP'

```

- Archivo “CARGAR_DATOS para FCP”

```

PPar=[[49.108,49.851,49.981,49.457,49.997],
[47.656,49.177,49.603,47.85,49.783],
[17.709,30.827,34.347,20.238,37.037],
[46.47.453,47.799,45.874,47.889],
[47.864,47.999,48,47.961,48],]

PPcl=[[104.46,104.46,104.46,104.46,103.51],
[266.33,266.33,266.33,266.33,266.33
[69.91,69.91,69.91,69.91,67.427],
[12.069,17.958,4.4005,5.2065,0.53908],]

```

```

PPcm=[[0,0,0,0,0],
[100.85,74.451,375.62,186.25,337.99],]

PPct=[[2.6298,5.6307,7.9868,12.331,10.901],
[3.2867,0.35273,2.676,1.2311,7.7658],
[44.632,11.045,39.013,12.955,64.944],
[6.0637,8.8269,19.231,6.4842,25.122],]

PPgln=[[9.3324e-007,0.0011206,0.041806,0.0071179,1.024e-005],
[0.39356,2.0016,6.4328,3.8398,0.79058],
[0.055696,0.95973,1.5598,0.72336,0.33864],
[1.6253,7.0254,8.8058,6.4481,4.0143],
[2.5871,7.5011,13.53,10.426,3.21091],]

PPgls=[[0.004535,1.58e-008,0.004411,0.15663,0.013937],
[5.9918,5.5029,5.8661,5.9987,5.9355],
[6,5.8966,6,6,5.9988],
[4.2103,0.00042299,3.3672,3.5124,0.39758],]

PPrn=[[69.505,62.753,97.495,86.822,98.504],
[121.49,143.8,184.51,172.69,182.63],
[7.9218,17.355,88.181,44.621,75.968],]

print"Los datos se han cargado, ejecute el programa: FCP_09"

```

A.2.5 Descripción detallada de código del programa “FCP”.

Seguidamente se describe el código del programa.

1º. Como se ha dicho anteriormente se van a generar potencias aleatorias para las cargas del sistema, mediante una FDP normal; para poder hacer esto en PSS/E con lenguaje Python es necesario importar a PSS/E el módulo de Python que permite generar números aleatorios de acuerdo a diferentes distribuciones de probabilidad. Por tanto se incluye al inicio del programa la instrucción:

```
import random
```

2º. Lo siguiente que ha sido necesario hacer es crear dos matrices con las dimensiones adecuadas para almacenar las cargas de las líneas y las tensiones de las subestaciones especificadas.

La matriz de cargas por las líneas tendrá tantas filas como líneas se quieran analizar y tantas columnas como número de ejecuciones de este programa sean necesarias. Para la matriz de tensiones en las subestaciones se sigue el mismo criterio:

```

CIRCULALIN=[[0 for c in range(NumEjec)] for f in range(len(BUSLIN)/3)]
# "f es el num. de filas y c en num. de columnas"

TENSIONNUDO=[[0 for C in range (NumEjec)] for F in range(len(BUSTENSION))]
# "F es el num. de filas y C en num. de columnas"

```

3º. Una vez creadas dichas matrices pasamos a describir el bucle principal de programación, que se ejecutará tantas veces como número de iteraciones haya especificado el usuario.

3.1 Lectura del caso base e inicialización de la variable “SECUNDARIA”.

En cada ejecución se comienza leyendo el caso de partida o caso base “.sav” e inicializando a cero la variable que almacenará la variación de generación eólica resultante.

Se ha llamado secundaria a esta variable porque la forma en que en este método se reparte la variación de generación eólica entre la generación térmica es muy parecida a como se hace en la realidad en el sistema eléctrico español y eso es la regulación secundaria.

```
psspy.case (CASOsav)
SECUNDARIA=0
```

3.2 Modificación de las potencias de los parques eólicos.

Lo siguiente que se hace (mediante bucles) es, para cada una de las 7 zonas de generación eólicas, extraer del caso de partida la potencia generada por cada parque eólico de la zona, comprobar si el parque está conectado y si su potencia generada es positiva, si no se cumple alguna de estas dos condiciones se fija la potencia del parque a cero.

Se introduce en el caso de partida, a cada parque, la potencia aleatoria calculada para la ejecución en curso.

Se resta a la potencia generada por el parque en el caso base, la potencia aleatoria calculada y se almacena la variación de potencia en la variable “SECUNDARIA”.

Se muestra el código de este bucle para una zona (Aragón):

```
# "Parques de la zona de ARAGON"
ng=0 # Contador del numero de bus de la zona, para la matriz de numeros de bus de los parques (BUSNUGar)"
nP=0 # Contador del numero de parque de la zona, para la matriz de potencias calculadas (PPar)"
while ng<len(BUSNUGar):
    # "De cada nudo lo primero es ver si esta conectado y si la potencia generada es positiva"
    st=psspy.macint(BUSNUGar[ng],BUSNUGar[ng+1],'STATUS')
    Pg=psspy.macdat(BUSNUGar[ng],BUSNUGar[ng+1],'P')
    P=Pg[1] # Pg es (0,potencia generada)
    if st[1]==0 or P<0:
        P=0
    # "Pone en cada nudo la potencia calculada por Matlab"
    psspy.machine_data_2(BUSNUGar[ng],BUSNUGar[ng+1],[_i],[PPar[nP][1]])

    # "Sumar las variaciones de generacion eolica para posteriormente repartirla entre diversos grupos termicos"
    SECUNDARIA=SECUNDARIA+(P-PPar[nP][1])
    ng=ng+2
    nP=nP+1
```

- Nótese que lo único que cambia en el código de los bucles, de una zona a otra, es el nombre de las variables: “BUSNUGar”, “PPar”

3.3 Modificación de la potencia consumida por las cargas.

En este caso, como se dijo anteriormente, se van a generar potencias aleatorias consumidas por las cargas, con una distribución de probabilidad normal.

Para hacer esto se utiliza un bucle de programación, que realizará el cálculo de una potencia aleatoria, para cada carga que se haya especificado en el archivo de datos de entrada.

Lo primero que se hace es comprobar si la carga está conectada o no:

- Si no está conectada, no se hace nada y se pasa a la siguiente carga.
- Si está conectada, se extrae, del caso base, la potencia que consume y se calcula la desviación típica de la potencia consumida, se ha considerado una variación de la potencia consumida por la carga, de un 10% (a partir de datos experimentales).

Con estos dos datos y mediante una FDP normal se calcula una potencia aleatoria consumida por la carga. En este caso la media de la distribución de probabilidad normal es la potencia consumida.

Al crear el código se ha utilizado una FDP gaussiana en vez de una normal, porque la generación de números aleatorios con esta última es más lenta que con una FDP gaussiana, en lenguaje Python y ambas FDP son idénticas.

Una vez obtenida la nueva potencia consumida por la carga, ésta se introduce en el caso base, como potencia consumida por dicha carga y se pasa a la siguiente carga.

Seguidamente se muestra el código:

```
# "Nudo de carga"
nc=0 # "Contador del numero de cargas"
while nc < len(BUSNUC):
    # "De cada nudo lo primero es ver si esta conectada, sino no hacemos nada"
    stc=psspy.lodint (BUSNUC[nc], BUSNUC[nc+1], 'STATUS')
    statusc=stc[1] # "Se pone stc[1] porque stc es (0,estado de la carga: 0 =desconectada y 1 =conectada)"
    if statusc==1:

        # "De cada nudo se extrae la potencia de la carga, Pc"
        Potmed=psspy.loddt2 (BUSNUC[nc], BUSNUC[nc+1], 'MVA', 'ACT')
        Pc=Potmed[1].real # "Saca la potencia como un num. Complejo, donde la parte real es la Pot activa
                        y la parte imaginaria es la Pot reactiva"
        desvtic=Pc*0.1 # "Se considera el 10% de la potencia activa de la carga"
        potaleat=random.gauss(Pc,desvtic) # "Genera una potencia aleatoria con distribucion Normal,
                                          la desviacion tipica es el 10% de Pc"

        # "Mete en el nudo la potencia aleatoria en MW, generada con la Normal"
        psspy.load_data (BUSNUC[nc], BUSNUC[nc+1], [_i], [potaleat])
    nc=nc+2
```


3.4 Asignación de la variación de potencia de la generación eólica entre diversos grupos térmicos del sistema eléctrico español.

En este método, la asignación de la variación de la potencia eólica surgida, se ha hecho equitativamente entre los generadores térmicos acoplados en el sistema; de forma que se comprueba que generadores están acoplados y se divide la variación de potencia eólica entre el número de generadores acoplados; asignando a cada generador térmico, la potencia que debe aumentar o disminuir. De esta forma al converger el caso, una vez modificada toda la generación eólica, el nudo oscilante no tiene que absorber las variaciones de potencia surgidas. Resulta de esta forma una simulación más acorde a la realidad.

Seguidamente se muestra el código:

```
#"Reparte la variaciones de generacion eolica entre diversos grupos termicos del sistema"
nt=0 # "Contador del numero de generadores termicos"
numacoplados=0 # "Numero de generadores termicos acoplados"
while nt < len(BUSTERMICA):
    staterm=psspy.macint (BUSTERMICA[nt], BUSTERMICA[nt+1], 'STATUS')
    statusterm=staterm[1]
    if statusterm==1:
        numacoplados=numacoplados+1

    nt=nt+2

SECUNINDIVIDUAL=SECUNDARIA/numacoplados # "Se reparte entre los grupos termicos acoplados la secundaria
                                          del sistema,a partes iguales"
nt=0 # "Contador del numero de generadores termicos"
while nt < len(BUSTERMICA):
    staterm=psspy.macint (BUSTERMICA[nt], BUSTERMICA[nt+1], 'STATUS')
    if staterm[1]==1:
        Pgt=psspy.macdat (BUSTERMICA[nt], BUSTERMICA[nt+1], 'P')
        Ptnueva= Pgt[1] + SECUNINDIVIDUAL
        psspy.machine_data_2(BUSTERMICA[nt], BUSTERMICA[nt+1], [_i], [Ptnueva])

    nt=nt+2
```

3.5 Realización de un flujo de cargas.

Una vez finalizado el proceso de modificación de las potencias de los parques eólicos y de las cargas del caso base y de repartición de la variación de potencia surgida, entre los grupos térmicos, es necesario converger el caso con el nuevo escenario resultante, para obtener los nuevos valores de circulación por la líneas y las tensiones en las subestaciones.

La convergencia del caso se realiza mediante un flujo de cargas en alterna, realizado con el método Newton-Raphson. De los posibles métodos que se pueden elegir en PSS/E para converger un caso, se ha seleccionado éste porque aporta unos resultados bastante fiables y el tiempo de cálculo es reducido.

La instrucción para realizar dicho flujo de cargas es:

```
psspy.fdns ([0,0,0,1,1,0,99,0])
```

3.6 Almacenamiento de las circulaciones por las líneas y de las tensiones en las subestaciones especificadas.

Una vez realizado el flujo de cargas, se leen del caso ya convergido, mediante 2 bucles de programación, los nuevos valores de circulación por las líneas y de tensión en las subestaciones especificadas; estos valores se almacenan en las matrices creadas al inicio del programa, “CIRCULALIN” y “TENSIONNUDO”.

En cada fila de las matrices se encuentran los valores de cada línea o nudo y cada columna corresponde al número de ejecución del bucle principal.

Se muestra el código que realiza dicho almacenamiento:

```
# " Se extraen los valores de circulacion de las líneas y se almacenan"
nl=0 # "Contador de numero de líneas"
while nl < len (BUSLIN):
    Pl=psspy.brnmisc (BUSLIN[nl], BUSLIN[nl+1], BUSLIN[nl+2], 'MVA')
    CIRCULALIN [nl/3] [l] = Pl[1] # "nl/3" son las filas y "l" las columnas. Ponemos Pl[1] porque
                                # Pl es (0, potencia por la línea en MW)
    nl=nl+3

# "Se extraen los valores de las tensiones en los nudos y se almacenan"
nb=0 # "Contador del numero de nudos de los que queremos la tension"
while nb < len (BUSTENSION):
    tensKV=psspy.busdat (BUSTENSION [nb], 'KV')
    TENSIONNUDO [nb] [l]= tensKV [1] # "Ponemos tensKV[1] porque tensKV es (0,tension del nudo)"
    nb=nb+1
```

- Hasta aquí se han descrito las diferentes partes y el funcionamiento del bucle principal del programa “FCP”. Tal como se indicó anteriormente este proceso se repetirá el número de veces mínimo que se haya determinado (1000,5000, ...), basado en el método de Montecarlo, que es el utilizado en este caso para realizar el flujo de cargas probabilístico.

Sólo queda añadir en el código del programa la instrucción para que, llegado este punto comience una nueva ejecución del bucle: `l=l+1`

4º. Una vez realizadas las ejecuciones necesarias, del bucle principal, se realiza el tratamiento de los resultados y la presentación en los archivos de salida.

A continuación se describe el proceso:

- Archivo de salida con las circulaciones de las líneas especificadas.

Para analizar los resultados obtenidos se realizará un análisis en frecuencia de las circulaciones de las líneas y se mostrarán las circulaciones de las líneas que superen el porcentaje de carga especificado, “UMBRAL”.

Para realizar este análisis en frecuencia primeramente se define una matriz en la que se almacenarán los intervalos entre los que se encuentran las circulaciones de cada línea a vigilar. Esta matriz de intervalos “subintervalosLIN” tendrá tantas filas como número de líneas a vigilar y tantas columnas como número de intervalos, en los que se deseen repartir las circulaciones de cada línea, más una.

De manera que si para una línea se han obtenido circulaciones entre 210 y 220 MW y se quieren repartir las circulaciones en 5 intervalos, esta matriz tendrá 6 columnas, conteniendo cada una de ellas el valor límite del intervalo. P.e. [210, 212, 214, 216, 218, 220]

Se muestra el código para crear dicha matriz:

```
subintervalosLIN=[[0 for c in range (numedintervalos+1)] for f in range (len (BUSLIN)/3)]
```

Una vez creada la matriz de intervalos, se extrae de la matriz de circulaciones de líneas, para cada línea, el valor máximo y mínimo de circulación obtenido y se calcula la amplitud de los intervalos. Con el valor mínimo y la amplitud de los intervalos se calculan los límites de los intervalos de circulaciones y se introducen estos en la matriz de intervalos.

Esto se realiza mediante 2 bucles anidados como se muestra a continuación:

```
CL=0
while CL<len (BUSLIN)/3:
    MIN=min (CIRCULALIN [CL])
    MAX=max (CIRCULALIN [CL])
    amplinterv=(MAX-MIN)/numedintervalos # "amplitud de los subintervalos"
    nsi=0
    while nsi<=numedintervalos:
        subintervalosLIN [CL][nsi]=MIN+(amplinterv * nsi) # "CL es la fila y nsi la columna"
        nsi=nsi+1
    CL=CL+1
```

Seguidamente se asignan las circulaciones obtenidas, para cada línea, al intervalo que les corresponda y se contabiliza el número de circulaciones que hay en cada intervalo.

Para ello es necesario crear una matriz en la que se almacenarán las circulaciones de las líneas clasificadas por intervalos. Esta matriz, “contadorLIN”, tendrá tantas filas como número de líneas a vigilar y tantas columnas como número de intervalos.

El código para crearla es:

```
contadorLIN=[[0 for c in range(numedintervalos)] for f in range(len(BUSLIN)/3)]
```

Para clasificar las circulaciones de cada línea, asignándolas al intervalo que les corresponda y realizar el recuento del número de circulaciones que resultan en cada intervalo; se utilizan cuatro bucles anidados, de forma que línea por línea se va recorriendo la fila que le corresponda de la matriz de circulaciones y se asigna cada valor de carga al intervalo que le corresponda, realizando el recuento del número de circulaciones que contiene cada intervalo.

El código del proceso es el siguiente:

```
CL=0
while CL<len(BUSLIN)/3:
    i=0
    while i<len(CIRCULALIN[CL]):
        nsi=0
        while nsi<numedintervalos:
            if CIRCULALIN [CL] [i] >= subintervalosLIN [CL] [nsi]
            and CIRCULALIN [CL] [i] <= subintervalosLIN [CL] [(nsi)+1]:
                contadorLIN[CL][nsi]=1+contadorLIN[CL][nsi]
                i=i+1
                break
            else:
                nsi=nsi+1
        CL=CL+1
```

Lo que queda ahora es escribir los resultados en un archivo de texto, para que el usuario pueda analizarlos.

Para ello lo primero que se ha de hacer es crear el archivo de texto y abrirlo en modo escritura. La ruta y nombre del archivo fue especificada por el usuario en el archivo de datos de entrada y se asignó a la variable "SALIDALINEA".

El código que realiza esto es:

```
f=open(SALIDALINEA,"w")
```

Se comienza escribiendo el título del contenido del archivo.

```
f.write("\n"+"      RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS \n\n")
```

Con (\n) se consigue escribir en la línea siguiente.

Después se extrae del caso con el que se está trabajando, línea a línea mediante un bucle, el ratio de carga de cada línea a vigilar y se comprueba si las distintas circulaciones obtenidas para las líneas, superan el porcentaje de carga establecido. Cada circulación de la línea que supere el valor de carga fijado se almacena en el vector creado para ello "CIRCULALINSOBRECARG=[]". Este vector se inicializa cada vez que se analizan las circulaciones de una línea y el número de elementos que contenga, va a servir para determinar la probabilidad de superar el valor de carga fijado.

Para poder interpretar los resultados es necesario escribir el nombre de cada línea y eso se consigue extrayendo del caso, los nombre de los subestaciones que unen.

Las instrucciones para ello son:

```
BUSPRI = psspy.notona (BUSLIN [nl] )  
BUSSEG = psspy.notona (BUSLIN [nl+1] )
```

Se calcula la probabilidad, en %, de que la línea supere el valor de carga fijado; como el número de veces que la carga de la línea ha superado el porcentaje de su carga fijado entre el número de iteraciones del flujo de cargas probabilístico.

```
veces = len (CIRCULALINSOBRECARG)  
probabilidad = veces*100/ NumEjec
```

Una vez hecho esto es el momento de escribir los resultados en el archivo de texto. Se escribe el nombre de la línea, el nivel de tensión, el número de circuito, el porcentaje de carga del que se quiere conocer la probabilidad de superarlo y dicha probabilidad.

También se añaden, para cada línea, las circulaciones que superan ese porcentaje de carga.

El código que realiza todo lo descrito anteriormente es el siguiente:

```
f=open(SALIDALINEA,"w")  
f.write("\n"+"      ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS      \n\n\n"+  
      "Ruta y nombre del caso de estudio: "+ str(CASOsav)+"\n\n\n"+  
      " ----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS ----- \n\n")  
nl=0  
FL=0  
CIRCULALINSOBRECARGADAS=[]
```

```

while nl<len(BUSLIN):
    col=0
    CIRCULALINSOBRECARG=[] # "Creamos un vector para almacenar las circulaciones de cada linea, por encima
                             del nivel de carga elegido"
    ratio=psspy.brndat (BUSLIN [nl], BUSLIN [nl+1],BUSLIN [nl+2], 'RATEA')
    while col < len (CIRCULALIN [FL]):
        if CIRCULALIN [FL] [col] > (ratio[1] * UMBRAL/100):
            CIRCULALINSOBRECARG.append (CIRCULALIN [FL] [col])
            col=col+1
        else:
            col=col+1
    BUSPRI=psspy.notona (BUSLIN [nl])
    BUSSEG=psspy.notona (BUSLIN [nl+1])
    veces=len (CIRCULALINSOBRECARG)
    probabilidad=veces*100/NumEjec
    f.write ("La linea: "+str(BUSPRI[1])+" - "+str(BUSSEG[1])+", ckt: "+str(BUSLIN[nl+2])+", cargara por encima del: "+
    str (UMBRAL)+" %, con una probabilidad del: "+str (probabilidad)+" %"+"\\n\\n"+
    "Circulaciones de la linea por encima del "+str (UMBRAL)+" % de su capacidad: [")
    e=0
    while e<len(CIRCULALINSOBRECARG):
        f.write ("%2f " % (CIRCULALINSOBRECARG [e])+", ")
        e=e+1
    f.write("]" + "\\n\\n")
    FL=FL+1
    nl=nl+3

```

- Como información adicional en este archivo de salida, se muestra para cada línea el análisis en frecuencia de las circulaciones obtenidas.

Mediante 2 bucles anidados, se escribe en este archivo de texto el nombre de cada línea y los intervalos en los que se han dividido las circulaciones obtenidas, así como el número de veces que la circulación resultó dentro de cada intervalo.

Al acabar de escribir en el archivo es necesario cerrarlo, “f.close()”.

Se muestra el código:

```

f.write("\\n\\n"+"-----ANALISIS EN FRECUENCIA DE LAS CIRCULACIONES DE LAS LINEAS-----\\n\\n")

CL=0
nl=0
while CL < len(BUSLIN)/3:
    BUSPRI=psspy.notona (BUSLIN [nl] )
    BUSSEG=psspy.notona (BUSLIN [nl+1] )
    f.write("\\n"+"ANALISIS EN FRECUENCIA DE LA CIRCULACION DE LA LINEA: "+str (BUSPRI [1]) + " - "
    + str (BUSSEG [1]) + ", circuito: " +str (BUSLIN [nl+2] ) + "\\n\\n")
    nsi=0
    while nsi < numedintervalos:
        f.write ("Numero de veces que la carga de la linea estara en el intervalo: " +
        "%2f"% (subintervalosLIN [CL] [nsi]) + "- " + "%2f"% (subintervalosLIN [CL] [(nsi)+1]) +
        " : " + str(contadorLIN [CL] [nsi] + "\\n"))
        nsi=nsi+1
    CL=CL+1
    nl=nl+3

f.close()

```

- Archivo de salida con las tensiones en las subestaciones especificados.

En el caso de las tensiones en las subestaciones, se procede exactamente igual que con las circulaciones de las líneas.

Se construyen, para cada nudo, un número de intervalos “numedintervalos”, en los cuales están las tensiones obtenidas para cada subestación.

Se asignan las tensiones de cada nudo, al intervalo que les corresponda y se contabiliza el número de veces que la tensión del nudo ha estado comprendida en cada intervalo.

Se muestra el código que realiza esto:

```
# "CALCULO DE LOS SUBINTERVALOS DE TENSIONES"
intervalosTEN = [[0 for c in range (numedintervalos+1)] for f in range(len(BUSTENSION))]
FL=0
while FL < len(BUSTENSION):
    MIN = min (TENSIONNUDO [FL] )
    MAX = max (TENSIONNUDO [FL] )
    amplinterv = (MAX-MIN)/numedintervalos # amplitud de los intervalos
    nsi=0
    while nsi <= numedintervalos:
        intervalosTEN [FL] [nsi] = MIN + (amplinterv * nsi) # "FL es la fila y nsi la columna"
        nsi=nsi+1
    FL=FL+1

'-----'
# "REPARTE LAS TENSIONES EN LOS INTERVALOS CORRESPONDIENTES Y CONTABILIZA EL NUMERO DE
TENSIONES EN CADA INTERVALO"

contadorTEN= [[0 for c in range (numedintervalos)] for f in range (len (BUSTENSION))]
FL=0
while FL < len (BUSTENSION):
    i=0
    while i < len (TENSIONNUDO [FL]):
        nsi=0
        while nsi < numedintervalos:
            if TENSIONNUDO [FL] [i] >= intervalosTEN [FL] [nsi] and
            TENSIONNUDO [FL] [i] <= intervalosTEN [FL] [(nsi) +1 ]:
                contadorTEN[FL][nsi]=1+contadorTEN[FL][nsi]
                i=i+1
                break
            else:
                nsi=nsi+1
        FL=FL+1
```

Una vez clasificadas las tensiones obtenidas, en los diferentes intervalos; sólo queda escribir los resultados en el archivo de salida para las tensiones.

Al igual que en el caso de las líneas, lo primero es crear y abrir en modo escritura el archivo de texto. La ruta y nombre se especificaron en el archivo de datos de entrada.

Después se escribe el título y contenido del archivo y se muestra para cada subestación el análisis en frecuencia de las tensiones obtenidas.

Mediante 2 bucles anidados, se escribe el nombre de cada subestación, el nivel de tensión y los intervalos en los que se han dividido las tensiones obtenidas, así como el número de veces que la tensión de la subestación resultó dentro de cada intervalo.

Al finalizar la escritura se cierra el archivo.

Se muestra el código que realiza lo anterior:

```

t=open(SALIDATENSION,"w")

t.write("\n"+"          ARCHIVO DE SALIDA PARA LAS TENSIONES DE LAS SUBESTACIONES      \n\n\n"+
      "Ruta y nombre del caso de estudio:  "+ str(CASOsav)+"\n\n\n"+
      "-----ANALISIS EN FRECUENCIA DE LAS TENSIONES DE LAS SUBESTACIONES ----- \n")
FL=0
nt=0
while FL < len (BUSTENSION):
    BUSTEN = psspy.notona (BUSTENSION [nt] )
    t.write("\n"+"ANALISIS EN FRECUENCIA DE LA TENSION DEL NUDO: " + str(BUSTEN [1]) + "\n\n")
    nsi=0
    while nsi < numedintervalos:
        t.write ("Numero de veces que la tension del nudo estara en el intervalo: " + "%.3f" (intervalosTEN [FL] [nsi]) +
            " - " + "%.3f" (intervalosTEN [FL] [(nsi)+1]) + ": " +str (contadorTEN [FL] [nsi]) + "\n")
        nsi=nsi+1
    FL=FL+1
    nt=nt+1

t.close()

```

5º. Para finalizar el código del programa, se añade el siguiente mensaje de información al usuario, que aparecerá cuando acabe de ejecutarse el programa:

El flujo de cargas probabilístico ha finalizado, los resultados pueden verse en:

F:\Proyecto USB\Prog_FCP_2009\ SALIDALIN_fecha_hora.txt

F:\Proyecto USB\Prog_FCP_2009\ SALIDATEN_fecha_hora.txt

Los archivos de salida, con los resultados, tienen el siguiente aspecto.

```

          ARCHIVO DE SALIDA PARA LAS CIRCULACIONES DE LAS LINEAS

Ruta y nombre del caso de estudio:  F:\Proyecto USB\Prog_FCP_2009\20090421_1336.sav

----- RESULTADOS DEL ANALISIS DE CIRCULACIONES DE LAS LINEAS -----

La línea: TAFALL 1  220.00 - ORCOYE 1  220.00, ckt: 1, cargará por encima del: 90 %, con una probabilidad del: 60 %
Circulaciones de la línea, en MW, por encima del 90 % de su capacidad: [397.64 ,422.25 ,423.83 ,414.32 ,411.80 ,409.35 ,]

La línea: CORDOV 1  220.00 - SANGUE 1  220.00, ckt: 1, cargará por encima del: 90 %, con una probabilidad del: 0 %
Circulaciones de la línea, en MW, por encima del 90 % de su capacidad: []

La línea: MESON 2  220.00 - VIMIAN 1  220.00, ckt: 1, cargará por encima del: 90 %, con una probabilidad del: 0 %
Circulaciones de la línea, en MW, por encima del 90 % de su capacidad: []

La línea: TUDELA 1  220.00 - MAGALL 3  220.00, ckt: 1, cargará por encima del: 90 %, con una probabilidad del: 0 %
Circulaciones de la línea, en MW, por encima del 90 % de su capacidad: []

Continúa en la página siguiente

```

-----ANALISIS EN FRECUENCIA DE LAS CIRCULACIONES DE LAS LINEAS-----		
ANALISIS EN FRECUENCIA DE LA CIRCULACION DE LA LINEA:	TAFALL 1	220.00 - ORCOYE 1 220.00, circuito: 1
Numero de veces que la carga de la línea estará en el intervalo:		
	381.64 MW - 388.67 MW:	2
	388.67 MW - 395.70 MW:	1
	395.70 MW - 402.73 MW:	1
	402.73 MW - 409.76 MW:	1
	409.76 MW - 416.80 MW:	2
	416.80 MW - 423.83 MW:	2
ANALISIS EN FRECUENCIA DE LA CIRCULACION DE LA LINEA:	CORDOV 1	220.00 - SANGUE 1 220.00, circuito: 1
Numero de veces que la carga de la línea estará en el intervalo:		
	188.32 MW - 188.32 MW:	1
	188.32 MW - 188.33 MW:	0
	188.33 MW - 188.33 MW:	1
	188.33 MW - 188.34 MW:	3
	188.34 MW - 188.34 MW:	3
	188.34 MW - 188.34 MW:	1
ANALISIS EN FRECUENCIA DE LA CIRCULACION DE LA LINEA:	MESON 2	220.00 - VIMIAN 1 220.00, circuito: 1
Numero de veces que la carga de la línea estará en el intervalo:		
	28.02 MW - 38.54 MW:	3
	49.05 MW - 59.56 MW:	1
	59.56 MW - 70.08 MW:	1
	70.08 MW - 80.59 MW:	2
	101.62 MW - 112.13 MW:	1
	112.13 MW - 122.65 MW:	1
ANALISIS EN FRECUENCIA DE LA CIRCULACION DE LA LINEA:	TUDELA 1	220.00 - MAGALL 3 220.00, circuito: 1
Numero de veces que la carga de la línea estará en el intervalo:		
	10.57 MW - 13.14 MW:	1
	13.14 MW - 15.72 MW:	4
	15.72 MW - 18.29 MW:	1
	18.29 MW - 20.86 MW:	2
	20.86 MW - 23.43 MW:	1
	33.71 MW - 36.28 MW:	1

ARCHIVO DE SALIDA PARA LAS TENSIONES DE LAS SUBESTACIONES		
Ruta y nombre del caso de estudio: F:\Proyecto USB\Prog_FCP_2009\20090421_1336.sav		
----- ANALISIS EN FRECUENCIA DE LAS TENSIONES DE LAS SUBESTACIONES -----		
ANALISIS EN FRECUENCIA DE LA TENSION DE LA SUBESTACION:	P.G.RO 1	400.00
Numero de veces que la tensión de la subestación estará en el intervalo:		
	417.483 kV - 417.520 kV:	3
	417.520 kV - 417.558 kV:	2
	417.558 kV - 417.596 kV:	1
	417.596 kV - 417.633 kV:	1
	417.746 kV - 417.784 kV:	1
	417.784 kV - 417.821 kV:	1
ANALISIS EN FRECUENCIA DE LA TENSION DE LA SUBESTACION:	MESON 2	220.00
Numero de veces que la tensión de la subestación estará en el intervalo:		
	225.819 kV - 225.827 kV:	3
	225.836 kV - 225.844 kV:	1
	225.844 kV - 225.853 kV:	1
	225.853 kV - 225.861 kV:	1
	225.861 kV - 225.870 kV:	1
	225.895 kV - 225.903 kV:	3

ANALISIS EN FRECUENCIA DE LA TENSION DE LA SUBESTACION: TAFALL 1 220.00

Numero de veces que la tensión de la subestación estará en el intervalo:

229.148 kV - 229.236 kV: 2
229.236 kV - 229.325 kV: 2
229.325 kV - 229.413 kV: 2
229.413 kV - 229.501 kV: 2
229.501 kV - 229.589 kV: 1
229.942 kV - 230.030 kV: 1

ANALISIS EN FRECUENCIA DE LA TENSION DE LA SUBESTACION: MONTET 1 220.00

Numero de veces que la tensión de la subestación estará en el intervalo:

229.518 kV - 229.525 kV: 2
229.553 kV - 229.560 kV: 1
229.560 kV - 229.566 kV: 1
229.566 kV - 229.573 kV: 2
229.573 kV - 229.580 kV: 1
229.580 kV - 229.587 kV: 3

Referencias bibliográficas.

- [1] Julio Usaola; *Probabilistic load flow in systems with high wind power penetration*. Universidad Carlos III de Madrid, Agosto 22, 2008
Disponible en Web: <http://hdl.handle.net/10016/2845>
- [2] Juan A. Martínez Velasco y Jacinto Martín Arrendó; *Aplicación del ATP en el cálculo de flujo de cargas probabilístico empleando recursos de red*. Revista Iberoamericana del ATP, vol.4, Numero 2, Julio 2002.
Disponible en Web: <http://www.iitree-unlp.org.ar/caue/PapersRevista/2002-N2-FlujoProb.pdf>
- [3] Guido van Rossum. *Guía de aprendizaje de Python*. Fred L. Drake, Jr., editor. Release 2.4.1a0. Septiembre 2005.
Disponible en Web: <http://es.tldp.org/Tutoriales/Python/tut.pdf>
- [4] Manual de PSS/E v.31
- [5] Manual de Matlab v. 7.0
Disponible en Web: http://www.sisoft.ucm.es/Manuales/MATLAB_r2006b.pdf
- [6] Adolfo Guillermo Aguilar Fernández; *Análisis y mejora de flujos de materia prima, material en proceso y producto terminado para la empresa de inyección de plásticos Peguform México*. Cholula , Puebla, México 18 de Mayo de 2004, Capítulo 5.
Disponible en Web: http://catarina.udlap.mx/u_dl_a/tales/documentos/lii/aguilar_f_ag/capitulo5.pdf
- [7] Alberto Carbajo Josa, Tomás Domínguez Autrán y Salvador Fernández Arangas; *Situación y retos de las energías renovables*. Málaga, Abril 2009.
Disponible en Web: http://www.ree.es/publicaciones/pdf/situacion_retos_energias_renovables.pdf
- [8] Red Eléctrica de España; *Detalle de la estructura de generación en tiempo real*.
Disponible en Web: http://www.ree.es/operacion/curvas_demanda.asp
- [9] Red Eléctrica de España; *Casos PSS/E utilizados en PDBF*. Publicaciones.
Disponible en Web: <http://www.esios.ree.es/web-publica/>
- [10] Distribución Beta y Normal.
Disponible en Web: <http://es.wikipedia.org/wiki/>
- [11] Gerardo González Morales; *Previsión de las reservas y limitaciones de la red*. Octubre de 2006.
Disponible en Web: http://powwow.risoe.dk/publ/REE-forecastingImbalances_Delft_2006.pdf